NASA Contractor Report 4480

# Numerical Modeling Tools for Chemical Vapor Deposition

Thomas J. Jasinski and Edward P. Childs
*Creare Inc.*
*Hanover, New Hampshire*

Prepared for
Langley Research Center
under Contract NAS1-18648

National Aeronautics and
Space Administration

Office of Management

Scientific and Technical
Information Program

**1992**

# EXECUTIVE SUMMARY

Chemical vapor deposition (CVD) is an essential and widely used process for the fabrication of thin solid films for electronic, optical and surface modification applications. The microgravity environment of earth orbit has the potential to improve the quality and increase the yield of thin films grown by CVD. Assessing and developing this potential through solely an empirical approach will be both costly and time consuming. Reliable and proven numerical modeling methods for CVD provide an important additional tool to more effectively foster the development of CVD in space.

This report describes the results of a Small Business Innovation Research (SBIR) program to develop general purpose numerical modeling tools for CVD. An existing computer program called FLUENT was the foundation for this effort. At the beginning of Phase I of the SBIR project, FLUENT could already simulate some of the phenomena encountered in CVD, e.g., diffusion of mass, momentum and energy. Phase I implemented prototype enhancements which are most important for CVD:

1. Diffusion of an arbitrary number of chemical species.
2. Multiple chemical reactions.
3. Reactions on a solid surface.
4. Deposition on a solid surface.
5. Reaction rate expressions which are user-specified.

During Phase II, these enhancements were incorporated into current versions of FLUENT. Additionally, other physical models important for CVD were implemented. These are:

1. Full multicomponent diffusive transport of species as opposed to Fick's Law (which is rigorous only for binary or dilute mixtures).
2. Composition dependent transport properties based on ideal gas relations, or the provision to utilize user-specified mixture property relationships.
3. Time-varying gravity field or motion of the reactor vessel.
4. Thermal (Soret) diffusion of chemical species.
5. Thermophoretic force on particles.
6. "Body-Fitted", i.e., curvilinear, coordinates.

Many of the Phase I and Phase II physical models have already been incorporated into the currently released version of the FLUENT software, FLUENT V3.03. (The FLUENT software is licensed by Creare's sister company Fluent Inc.) The full complement of models will be incorporated into FLUENT V4.2.

As described in the paragraph above, the CVD modeling enhancements have either already or will be commercialized. In addition, the new modeling capabilities have been and are being used to support industrial CVD equipment manufacturers as part of Creare's contract services. Judged by the Phase III activity already in place, this SBIR program will provide important benefits to the CVD community.

# FOREWORD

This report was prepared by Creare Inc., Hanover, New Hampshire, under NASA contract NAS1-18648. This is the final technical report for this contract.

The work performed under this contract addresses Topic 15.06 of the NASA Small Business Innovation Research (SBIR) program solicitation 86-1. Topic 15.06 concerns development of computer software for realistic modeling of CVD processes. The work was administered by NASA Langley Research Center. Dr. Ivan Clark was the technical contact at NASA Langley.

This report covers work performed between May 5, 1988 and May 31, 1991. The Principal Investigator was Dr. Thomas Jasinski.

# Contents

# List of Figures

# List of Tables

# 1 INTRODUCTION

Chemical vapor deposition (CVD) is an essential and widely used process for the fabrication of thin solid films for electronic, optical and surface modification applications. The technology of CVD has developed largely through empiricism and the cumulative experience of the workers in the field. As advances in technology develop at an ever increasing rate, however, a solely empirical approach is both costly and time consuming.

Significant progress has been made over the past two decades in numerical modeling of many aspects of fluid flow. During this period, the cost of performing a numerical simulation of a physical process has decreased by a factor of ten every eight years while the cost of designing and performing laboratory experiments continues to rise [1]. This explains the current trend toward the use of numerical simulation tools to aid in the development of hardware and processes. The overall result is lower cost and lower risk.

The advantages of numerical simulation are especially apparent for R&D of space-based processes. The cost and time involved in developing and performing an experiment in space are very much greater than for a similar experiment on earth. Thus, there is a strong need to substantiate the design of an experiment during hardware development and to implement data acquisition capabilities which maximize the knowledge gained from the experiment. An accurate and reliable numerical predictive capability provides an important additional tool to achieve these goals.

The microgravity environment of earth orbit has the potential to improve many of the various CVD processes. Assessing and developing this potential through an empirical approach will be, as described above, both costly and time consuming. There is, therefore, a need for reliable and proven modeling tools for chemical vapor deposition.

Prior to this work, numerical modeling tools for CVD were largely developed ad hoc by the materials scientist to meet the rather limited scope of a particular R&D program. In so doing, only those effects important to the system under study were included in the model. The applicability of the most "comprehensive" of these models was consequently restricted. Modeling tools of adequate sophistication for the accurate prediction of the general CVD process was the primary goal of this project.

This report describes the results of Phase II of an SBIR program to develop general purpose numerical modeling tools for chemical vapor deposition (CVD). The modeling tools predict flow process variables such as fluid velocities, temperatures, and chemical species distributions within a CVD reactor as well as deposition rates on a substrate. Optimized reactor and process designs and a better understanding of experimental results are anticipated benefits of such a predictive capability.

The report is organized into five principle sections. The next section reviews overall program objectives, and summarizes the status of the development effort at the end of Phase II. Section 3 documents the enhancements made to FLUENT which extend its modeling capabilities for CVD processes. Section 4 provides examples of CVD simulations which exercise some of the new modeling capabilities. Finally, Section 5 summarizes the principal conclusions of the Phase II work.

# 2 OBJECTIVES AND SUMMARY OF CURRENT STATUS

## 2.1 SBIR PROGRAM OBJECTIVES

The CVD process is governed by a complex interaction of transport phenomena (fluid flow, heat transfer, chemical species transport), chemistry, thermodynamics, process conditions (e.g., flow rates, heater temperatures) and component geometries. The process may be either steady or time varying, laminar or turbulent, and may have properties which are temperature, composition and possibly pressure dependent. Predictive methods must include all these effects for accurate, general purpose modeling of CVD processes.

At the start of this SBIR program Creare developed and marketed a powerful computer program called FLUENT which could simulate a number of the phenomena encountered in CVD. (In 1989, development and marketing of the FLUENT software was undertaken by Fluent Inc., a spin-off company of Creare Inc.) This SBIR program addressed the development of an enhanced version of FLUENT specifically tailored for CVD processes. Thus, the starting software had well tested and verified physical models and we could therefore concentrate on the addition of new physical models of special importance for CVD.

In particular, the CVD version of FLUENT should be able to simulate the fluid flow, heat transfer and chemical reaction phenomena occurring during CVD with realistic geometries and boundary conditions. Prediction of temperature, velocity and mass composition distributions in three spatial dimensions as well as surface deposition rates, are required. Both gas-phase and surface reactions need to be simulated. Time varying gravity vectors should be included to permit simulation of space-based processing. Results should be presented in either graphical or tabular form. The menu-driven user interfacing must be extended commensurate with the additional features provided by the enhancements.

An important objective of the SBIR program is Phase III which includes marketing, support and continued development. As described in Section 2.4, Phase III activity began early during the Phase II project with the release of some of the CVD modeling capabilities in the then-current version of FLUENT. At present, Fluent Inc. plans to implement all of the CVD modeling enhancements developed during this SBIR program in FLUENT V4.2. Thus, the CVD community is assured that the CVD modeling tools developed in this project are treated in a manner consistent with other software products, including verification testing, documentation, user seminars, and portability between different computers.

## 2.2 PHASE I OBJECTIVES AND RESULTS

The objective of Phase I was to demonstrate that FLUENT provided a suitable foundation for the development of a general purpose modeling tool for CVD. The approach followed in Phase I was to enhance the then current version of FLUENT, Version 2.93, to include the most important CVD phenomena. The specific enhancements to be addressed in Phase I were:

- diffusion of more than three chemical species
- multiple chemical reactions
- chemical reactions at surfaces
- mass deposition on surfaces
- user-defined reaction rates

A new version of FLUENT, based on Version 2.93 as the starting point, was developed with the enhancements listed above. This new version of FLUENT was called FLUENT/CVD. FLUENT/CVD was demonstration tested on a sample CVD process: silicon deposition from silane in a horizontal reactor configuration. This system and configuration has a relatively large amount of experimental and numerical simulation data available in the literature. FLUENT/CVD correctly predicted qualitative trends and compared well with experimental data and other numerical predictions [2]. These favorable results satisfied the principal Phase I objective, i.e., FLUENT was suitable for realistic CVD simulation. The Phase I software was the starting point for the Phase II project.

FLUENT/CVD was delivered to NASA/Langley at the conclusion of Phase I.

## 2.3 PHASE II OBJECTIVES AND APPROACH

The overall objective of Phase II was to develop, demonstrate and deliver to NASA/Langley a general purpose CVD modeling tool. In addition, the computer software would be installed on a NASA/Langley computer so that NASA/Langley personnel would be able to use the codes during and at the completion of Phase II.

The specific technical objectives for Phase II were:

1. Develop, implement and test physical models not included in the Phase I program but required for general purpose CVD modeling. These included:
   - Composition dependent properties
   - Thermal diffusion
   - Time-varying gravity

- Thermophoresis of particles

2. Couple all CVD physical models developed in Phase I and II with capabilities for non-orthogonal geometry mapping. This would permit generalized CVD geometries to be more accurately modeled.

3. Develop a "smart" pre-processor for the non-orthogonal geometry version to simplify the input required for the generalized geometry description. In particular, this would be done for a NASA/Langley experimental CVD system.

4. Transfer modeling capabilities developed in Phase II as quickly as possible for easy assimilation by NASA/Langley personnel and begin as early as possible a program of verification against experimental results.

The approach to meeting the Phase II objectives was based on the development of two separate computer codes. The first computer code developed in Phase II would be based on FLUENT/CVD, the computer code developed in Phase I. FLUENT/CVD already had the principal chemical reaction and surface deposition models required for CVD. The additional physical models addressed in Phase II shown in item 1 above would therefore be implemented and tested in FLUENT/CVD. The Phase II enhanced version of FLUENT/CVD would have all the physical models for CVD addressed in Phases I and II. FLUENT/CVD, however, does not permit generalized non-orthogonal geometry modeling.

Another version of FLUENT existed at the start of Phase II: FLUENT/BFC (Body Fitted Coordinates). FLUENT/BFC permits accurate geometry modeling of systems with non-orthogonal boundaries. In contrast, FLUENT and FLUENT/CVD would require a "stair-stepped" grid to model boundaries which are not parallel to the coordinate axes. FLUENT/BFC, however, did not have many of the CVD simulation capabilities that FLUENT/CVD had at the start of Phase II.

The second item in the list of objectives above would be addressed by developing a second computer code based on FLUENT/BFC. FLUENT/BFC would be upgraded to include all of the CVD physical models addressed in both Phase I and Phase II of the SBIR program. This new version would be called FLUENT/BFC/CVD. The "smart" pre-processor (item 3 above) developed to simplify the geometry input for the NASA CVD reactor would be coupled to FLUENT/BFC/CVD.

## 2.4 SUMMARY OF PHASE II RESULTS AND STATUS

All of the CVD modeling capabilities included in the Phase II program except for one have been implemented in FLUENT V3.0. The lone modeling capability lacking in FLUENT V3.0 involves the species boundary condition, equation 3.54; which represents normal convection of species of surfaces, as occurs with deposition or etching. This effect is typically

small, however. The current release of FLUENT V3.0, Version 3.03, has been delivered to NASA/Langley.

FLUENT V3.0 handles orthogonal geometries only. As described in Section 2.3, all CVD modeling enhancements are to be also available in software which can handle curvilinear geometries. Toward this goal, work was started on enhancing FLUENT/BFC in accordance with the originally proposed workplan. During the course of Phase II, however, a new algorithm for curvilinear geometry mapping was developed which would allow both orthogonal and curvilinear mapping in the same code. With this capability, only one version of FLUENT, as opposed to two as before, would need to be developed. And for the user, only one version would need to be learned. For these reasons, FLUENT/BFC was scheduled for eventual termination from the FLUENT software family. A new version of FLUENT, Version 4, would include the curvilinear, as well as orthogonal, geometry mapping capabilities.

Continued implementation of CVD models in FLUENT/BFC would have resulted in a code delivered to NASA/Langley which would eventually be obsolete and unsupported. As a result, the workplan of Phase II was modified such that the CVD modeling capabilities would be provided in Version 4 of FLUENT rather than FLUENT/BFC. The "smart" preprocessor for a particular NASA/Langley reactor would, as well, be configured for Version 4.

FLUENT Version 4.10 was originally released during the fall of 1991 and was delivered to NASA/Langley. FLUENT V4.10 does not, however, include many of the CVD modeling capabilities. These are to be implemented during the second phase of Version 4 development and are scheduled for release as FLUENT V4.2 during the fall of 1992. NASA/Langley will also receive a copy of FLUENT V4.2. The "smart" preprocessor will be delivered with FLUENT V4.2.

# 3 PHYSICAL MODELS FOR CVD

Physical modeling enhancements for CVD incorporated into FLUENT during this SBIR project are described in this section. The enhancements are described within the format of providing the overall CVD modeling capabilities of the end-product software. For example, the entire conservation equation for fluid momentum is provided even though the principal enhancement involves only the body force term. Through this approach, it is intended that this section will serve as a concise and stand-alone reference for the physical models of the phenomena normally encountered in CVD simulation, eliminating the need for frequent reference to FLUENT documentation.

FLUENT incorporates physical models for many other phenomena than those within the scope of the Phase I and Phase II effort. For example:

- turbulent transport of momentum, energy and chemical species;
- porous fluid flow;
- fan model (step increase in pressure);
- heat exchanger model (step change in temperature);
- absorption/emission/scattering of thermal radiation in an optically thick gas;
- non-Newtonian fluids; and,
- others.

Although these models can be used as needed in any simulation, they are not normally of primary interest for CVD modeling. Description of these models is provided in the standard FLUENT documentation, e.g.,[3].

As described in Section 2.4, FLUENT V4.2 will be the ultimate repository of the CVD modeling tools developed in this project, both for orthogonal and non-orthogonal geometries. FLUENT V4.2 is scheduled for release by Fluent Inc. after the submission of this report. During the course of Phase II, however, versions of FLUENT with CVD modeling capabilities were provided to NASA/Langley, culminating in FLUENT V3.03 at the end of Phase II. In order to provide a more complete reference, the physical models described in this section will be based on the ultimate version of FLUENT, i.e., FLUENT V4.2. Distinctions will be made as required, between FLUENT V3.03 and FLUENT V4.2.

# 3.1 TRANSPORT EQUATIONS

## 3.1.1 Summary of Transport Equations

Table 3.1 lists, for reference, the conservation equations implemented in FLUENT. The sections which follow describe these equations in more detail including assumptions and restrictions which apply.

## 3.1.2 Conservation of Mass

The continuity equation expresses conservation of mass in the fluid:

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot \mathbf{G} \tag{3.1}$$

where $\rho$ is the fluid density and $\mathbf{G}$ is the mass flux vector. Equation 3.1 states that the rate of change of mass within an infinitesimal control volume in the fluid is equal to the net rate of convection of mass into the control volume across the control volume surfaces. It is assumed that there is no creation or destruction of mass. The mass flux vector, $\mathbf{G}$, is given by:

$$\mathbf{G} = \rho \mathbf{V} \tag{3.2}$$

Substituting equation 3.1 into equation 3.2 provides the working expression for conservation of mass:

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho \mathbf{V}) \tag{3.3}$$

where $\mathbf{V}$ is the vector velocity of the fluid.

There was no change to the continuity equation during the Phase I or Phase II projects.

## 3.1.3 Conservation of Momentum

Conservation of momentum in a fluid leads to the following expression in vector–tensor notation[4][5]:

$$\rho \frac{\partial \mathbf{V}}{\partial t} + (\mathbf{V} \cdot \nabla)\mathbf{V} = \nabla \cdot \tau + \sum_{i=1}^{n} \rho_i \mathbf{F_i} \tag{3.4}$$

where $\tau$ is the fluid stress tensor and $\mathbf{F_i}$ is the external force exerted upon species $i$ per unit mass of species $i$. The left hand side of equation 3.4 represents the rate of increase of fluid momentum in an infinitesimal control volume; the increase of momentum is either stored in the control volume (first term) or convected out of the control volume (second term). As Newton's Law demands, the rate of increase of momentum is equal to the net force on the control volume. The stress tensor term on the right hand side of equation 3.4 provides the

net force on the surfaces of the control volume while the second term on the right hand side provides the net body force. Equation 3.4 actually represents three separate equations—one for each coordinate direction.

The external body force, $\mathbf{F}_i$, results from several phenomena. The most commonly encountered body force is gravity. Gravity acts equally on each species in a fluid mixture. Another body force, for example, may be caused by exposing the fluid to an electrostatic field. In this case, the body force exists for only thoses species which have a non-zero electrical charge. At present, only species-independent body forces are implemented. Also, the body force may be a function of time but not of position. Under the above conditions, the last term in equation 3.4 becomes:

$$\sum_{i=1}^{n} \rho_i \mathbf{F_i} \longrightarrow \rho \mathbf{F}(t) \tag{3.5}$$

Stokes' laws of viscosity relate the stress tensor, $\tau$, to the physical variables of pressure and velocity. These relations can be found in many texts on fluid dynamics, e.g., [5][6]. The Navier-Stokes equations result when Stokes' laws of viscosity are substituted into equation 3.4. As an example, the x-direction Navier-Stokes equation in Cartesian coordinates is shown below.

$$\rho\frac{\partial u}{\partial t} + \rho\mathbf{V}\cdot\nabla u = -\frac{\partial p}{\partial x} + \frac{\partial}{\partial x}\left[\mu\left(2\frac{\partial u}{\partial x} - \frac{2}{3}\nabla\cdot\mathbf{V}\right)\right] + \frac{\partial}{\partial y}\left[\mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)\right] + \frac{\partial}{\partial z}\left[\mu\left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}\right)\right] + \rho F_x(t) \tag{3.6}$$

There are similar equations for the y and z direction in the Cartesian coordinate system as well as for the different coordinate directions in the cylindrical or spherical coordinate systems.

The Navier-Stokes equations, are implemented in FLUENT V4.2 in the Cartesian and cylindrical coordinate systems. Table 3.1 provides for reference the appropriate expressions in the Cartesian coordinate system.

A simplified form of the Navier-Stokes equations was implemented in FLUENT prior to the start of Phase II. The simplification resulted from assuming that the divergence of the velocity vector is zero:

$$\nabla\cdot\mathbf{V} = 0 \tag{3.7}$$

High Mach number flows (greater than about 0.3) can not be properly simulated with this restriction. For low Mach number flows (less than about 0.3) typical of most CVD processes, however, $\nabla\cdot\mathbf{V}$ is small compared to the other terms in the Navier-Stokes equations. In such cases, neglecting $\nabla\cdot\mathbf{V}$ introduces negligible error.

Removing the restriction of equation 3.7 was carried out under development of FLUENT version 3 (i.e., not funded by the Phase II program) after the first version of FLUENT/CVD was delivered to NASA Langley in August 1989. Therefore, this earlier version of FLUENT/CVD assumed that $\nabla\cdot\mathbf{V} = 0$.

Table 3.1: **Summary of Transport Equations**

| Conserved Quantity | Conservation Equation | Comments |
|---|---|---|
| **Mass** | $\frac{\partial \rho}{\partial t} = -[\frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z}]$ | • No creation/destruction of mass |
| **Vector Momentum** | x-Direction<br>$\rho\frac{Du}{Dt} = -\frac{\partial p}{\partial x} + \rho F_x + \frac{\partial}{\partial x}[\mu(2\frac{\partial u}{\partial x} - \frac{2}{3}\nabla \cdot \boldsymbol{V})]$<br>$+\frac{\partial}{\partial y}[\mu(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x})] + \frac{\partial}{\partial z}[\mu(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z})]$<br>y-Direction<br>$\rho\frac{Dv}{Dt} = -\frac{\partial p}{\partial y} + \rho F_y + \frac{\partial}{\partial y}[\mu(2\frac{\partial v}{\partial y} - \frac{2}{3}\nabla \cdot \boldsymbol{V})]$<br>$+\frac{\partial}{\partial x}[\mu(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x})] + \frac{\partial}{\partial z}[\mu(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y})]$<br>z-Direction<br>$\rho\frac{Dw}{Dt} = -\frac{\partial p}{\partial z} + \rho F_z + \frac{\partial}{\partial z}[\mu(2\frac{\partial w}{\partial z} - \frac{2}{3}\nabla \cdot \boldsymbol{V})]$<br>$+\frac{\partial}{\partial x}[\mu(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z})] + \frac{\partial}{\partial z}[\mu(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y})]$ | • $F = g(t) + a_{ni}(t)$<br>• Body force F species independent |
| **Chemical Species** | $\rho\frac{DX_i}{Dt} = -\nabla \cdot J_i + R_i$; i=1, n-1<br>$J_n = 1 - \sum_{i=1}^{n-1} J_i$<br>$J_i = \frac{C^2}{\rho}\sum_{j=1,\neq i}^{n} M_i M_j D_{ij}\nabla Y_j - D_i^T\frac{\nabla T}{T}$ | • Expression of $J_i$ assumes fluid mixture is an ideal gas<br>• $J_i$ for dilute mixtures ($Y_i << Y_n$):<br>$J_i = -\rho\mathcal{D}_{in}\nabla X_i - D_i^T\frac{\Delta T}{T}$ |
| **Energy** | $\rho\frac{Dh}{Dt} = -\nabla \cdot k\nabla T - \nabla \cdot (\sum_{i=1}^{n} h_i J_i) + \frac{Dp}{Dt} + \Phi$ | • Neglect volumetric heating<br>• Neglect work of external body forces |

Table 3.2: **Nomenclature for Table 3.1**

| Variables | Description |
|---|---|
| $a_{ni}$ | non-inertial acceleration of solution domain |
| $C$ | molar concentration |
| $D_{ij}$ | multicomponent mass diffusion coefficient |
| $D^T$ | thermal diffusion coefficient |
| $\mathbf{F}$ | body force vector per unit mass |
| $\mathbf{g}$ | gravity vector |
| $h$ | enthalpy |
| $\mathbf{J}$ | mass flux vector |
| $k$ | thermal conductivity |
| $M$ | molecular weight |
| $n$ | number of species in the fluid mixture |
| $p$ | pressure |
| $R$ | species generation rate per unit volume |
| $t$ | time |
| $T$ | temperature |
| $u$ | x-direction velocity |
| $v$ | y-direction velocity |
| $\boldsymbol{V}$ | vector velocity |
| $w$ | z-direction velocity |
| $x$ | coordinate direction |
| $X$ | mass fraction |
| $y$ | coordinate direction |
| $Y$ | mole fraction |
| $z$ | coordinate direction |
| | |
| GREEK | |
| $\rho$ | density |
| $\mu$ | viscosity |
| $\Phi$ | viscous dissipation, see Eq. 3.29 |
| | |
| SUBSCRIPTS | |
| $i, j$ | species designation |
| $n$ | currier gas species |
| $x, y, z$ | coordinate direction |
| | |
| OTHER | |
| $\mathcal{D}$ | binary diffusion coefficient |
| $\nabla$ | vector gradient operator |
| $\frac{D}{Dt}$ | $\equiv \frac{\partial}{\partial t} + u\frac{\partial}{\partial x} + v\frac{\partial}{\partial y} + w\frac{\partial}{\partial z}$ |

FLUENT V3.03 delivered at the end of the project and FLUENT V4.2, however, remove this restriction and the conservation of momentum equations are as shown in Table 3.1. The enhancements made to the momentum equations under the Phase II effort involve time-varying body forces to (1) simulate changing magnitude and direction of the gravity vector and (2) the effect of motion of the solution domain. These enhancements are described in the following sections.

#### 3.1.3.1 Time-Varying Gravity

The gravity component of the body force $\mathbf{F}(t)$ was enhanced during Phase II to allow each component of the gravity vector, $\mathbf{g}$, to vary with time. Prior to Phase II, the gravity vector was time invariant. Input menus have been implemented for setting time-varying values of $\mathbf{g}$ in any coordinate direction in Cartesian or cylindrical coordinates. A constant value of $\mathbf{g}$ is used for a steady-state model.

#### 3.1.3.2 Non-Inertial Frame of Reference

The modeling available at the end of Phase I assumed that the solution domain remains fixed to or travels at a constant velocity with respect to an inertial frame of reference. It may occur, however, that the physical boundaries of the flow domain (e.g., the confining walls) are not moving at a constant velocity with respect to an inertial reference. This would be the case, for example, for a CVD reactor rigidly attached to the structure of a space station when the space station rotates on its own axis and/or experiences structural vibration known as "g-jitter".

Simulation of fluid flow in a process for which the flow boundaries are non-inertial may be accomplished in two ways. First, the frame of reference in the numerical model may be a convenient inertial frame of reference. In this case, the boundaries of the flow domain must accelerate in a prescribed fashion with respect to the inertial frame of reference. Computationally, this requires a complex remapping of the physical flow domain on the inertial frame of reference at each successive time step. The second treatment considers that the frame of reference for the numerical model is non-inertial and fixed to the accelerating flow-domain. In this case, an "inertial" body force must be applied to each control volume in the flow domain. The inertial body force is related to the acceleration of the flow-domain as follows [5]:

$$\mathbf{f}_{ni}(t) = \rho(\mathbf{a}_{ni} + \propto_{\mathbf{ni}} \times \mathbf{r} + 2\omega \times \mathbf{V} + \omega \times (\omega \times \mathbf{r})) \tag{3.8}$$

where: $\mathbf{f}_{ni}(t)$ = body force per unit volume (vector) due to the calculational frame of reference being non-inertial

$\mathbf{a}_{ni}$ = linear acceleration (vector) of the calculational (non-inertial) frame of reference with respect to an inertial frame of reference

$\propto_{ni}$ = angular acceleration (vector) of the calculational (non-inertial) frame of reference with respect to an inertial frame of reference

$\omega$ = angular velocity (vector) of the calculational (non-inertial) frame of reference with respect to an inertial frame of reference

$\mathbf{r}$ = radius vector from the calculational (non-inertial) frame of reference to a point in the flow domain

$\mathbf{V}$ = fluid velocity in the calculational (non-inertial) frame of reference

The third and fourth terms within parentheses in equation 3.8 represent Coriolis and centrifugal forces, respectively. In general, each of $\mathbf{a}_{ni}$, $\propto_{ni}$, and $\omega$ may be functions of time.

The second approach for treating non-inertial flow domains described in the previous paragraph, i.e., that derived from equation 3.8, was chosen based on ease of implementation. Thus, $\mathbf{F}(t)$ in equation 3.5 needs only to include the contribution of $\mathbf{f}_{ni}(t)$ from equation 3.8. As a simple example, the total body force $\mathbf{F}(t)$ on the fluid in a non-inertial flow domain on the surface of the earth would be:

$$\mathbf{F}(t) = \rho\mathbf{g} + \mathbf{f}_{ni}(t) \tag{3.9}$$

where $\mathbf{g}$ is the gravitational acceleration vector.

The linear acceleration term of equation 3.8, i.e., $\boldsymbol{a}_{ni}(t)$, was implemented during Phase II. The time-varying vector components of $\mathbf{a}_{ni}$ are menu-prompted user inputs. Additionally, in FLUENT V4.2 the centrifugal and Coriolis force terms in equation 3.8 are available for simulation in the polar coordinate system where the axis of rotation is assumed to be the same as the axis of the coordinate system. Non-inertial centrifugal and Coriolis forces were developed separate from the Phase II program.

### 3.1.4 Conservation of Chemical Species

The conservation equation for each chemical species in the fluid mixture is [4]:

$$\rho\frac{\partial X_i}{\partial t} + \rho\mathbf{V}\cdot\nabla X_i = -\nabla\cdot\mathbf{J_i} + R_i \tag{3.10}$$

where: $X_i$ = mass fraction of species $i$

$\mathbf{J}_i$ = diffusional mass flux vector of species $i$

$R_i$ = net rate of creation of mass of species $i$ per unit volume

The first term on the left hand side of equation 3.10 represents the rate of increase of species $i$ in an infinitesimal control volume. The second term on the left hand side is the amount of

species $i$ convected out of the control volume per unit time. The first term on the right hand side represents the net rate of inflow of species $i$ by diffusion. The diffusional mass flux $\mathbf{J_i}$ is:

$$\mathbf{J_i} = \rho_i(\mathbf{V_i} - \mathbf{V}) \tag{3.11}$$

where: $\rho_i$ = mass concentration of species $i$ in the mixture
$\mathbf{V_i}$ = mean velocity, based on mass, of species $i$
$\mathbf{V}$ = mean velocity, based on mass, of the entire fluid mixture

The difference between $\mathbf{V_i}$ and $\mathbf{V}$ is therefore a measure of molecular mass diffusion. The last term in equation 3.10 represents the rate of increase of species $i$ in the control volume due to all chemical reactions. If there are $K$ chemical reactions, then:

$$R_i = \sum_{k=1}^{K} R_{ik} \tag{3.12}$$

where $R_{ik}$ is the rate of creation per unit volume of mass of species $i$ by reaction $k$.

Prior to Phase I, FLUENT could handle only up to three species conservation equations. During Phase I, the solution procedure of FLUENT/CVD was generalized to handle an arbitrary number of species. This solution procedure has been carried through into Phase II.

Each species in equation 3.10 introduces an unknown solution parameter $X_i$, the mass fraction of species $i$. However, the sum of all the mass fractions must equal unity:

$$\sum_{i=1}^{n} X_i = 1 \tag{3.13}$$

Equation 3.13 constrains the permissible values of $X_i$ and of the total of $n$ species in the fluid mixture, only $n-1$ are independent. As a consequence, only $n-1$ species transport equations are solved with the $n^{\text{th}}$ species concentration being determined from equation 3.13:

$$X_n = 1 - \sum_{i=1}^{n-1} X_i \tag{3.14}$$

Other unknowns in equation 3.10 are $\mathbf{J_i}$ and $R_i$. The value of $R_i$ depends on the values of $X_i$ and other solution variables (e.g., temperature, pressure) as described in Section 3.2. The value of $\mathbf{J_i}$ is related to the solution variables as described in the following sections.

#### 3.1.4.1 Diffusional Mass Flux–General Mixtures

A general expression for the diffusional mass flux vector of species i, i.e., $\mathbf{J}_i$, is given by [4]:

$$\mathbf{J}_i = \frac{C^2}{\rho} \sum_{j=1, j\neq i}^{n} M_i M_j D_{ij} \mathbf{d}_j - D_i^T \frac{\nabla T}{T} \tag{3.15}$$

where: C = molar concentration of the mixture
$M_i$ = molecular weight of species $i$
$M_j$ = molecular weight of species $j$
$D_{ij}$ = multicomponent diffusion coefficient for species pair $i$-$j$
$\mathbf{d}_j$ = mechanical diffusion vector for species $j$
$D_i^T$ = thermal mass diffusion coefficient for species $i$ in the mixture

The first term on the right hand side of equation 3.15 is the contribution to $\mathbf{J}_i$ due to concentration gradients while the second term represents the Soret effect—-mass transfer due to temperature gradients.

The mechanical diffusion vector is expressed as follows [4]:

$$\mathbf{d}_i = \frac{Y_i}{a_i}(\nabla a_i)_{P,T} + X_i Z(\rho\vartheta_i - 1)\frac{\nabla P}{P}$$

$$- \frac{X_i Z}{P}\left(\rho\mathbf{F}_i - \sum_{j=1}^{N}\rho_j\mathbf{F}_j\right) \tag{3.16}$$

where: $a_i$ = activity coefficient of species $i$ in the mixture
$Y_i$ = mole fraction of species $i$ in the mixture
Z = compressibility factor of the mixture, defined by the equation of state, Z=P/$C\bar{R}T$, where $\bar{R}$ is the universal gas constant
$\vartheta_i$ = partial specific volume of species $i$ in the mixture

The external body forces presently implemented are those due to gravity and to accelerating flow domains (see sections 3.1.3.1 and 3.1.3.2). In either case, the force $\mathbf{F}_i$ is the same for all species $i$ and the last term in equation 3.16 is identically zero. (This term would not disappear, for example, if one or more of the species had a net electrical charge and the flow domain was in an electrical or magnetic field.) Also, the pressure gradient term in equation 3.16 is neglected since it can be shown to be small for the small pressure gradients usually present in CVD reactors. With these restrictions, equation 3.16 simplifies to:

$$\mathbf{d}_i = \frac{Y_i}{a_i}(\nabla a_i)_{P,T} \tag{3.17}$$

### 3.1.4.2 Ideal Gas Approximation

Modeling of CVD using FLUENT assumes that the fluid mixture is an ideal gas. In this case, the activity coefficient, $a_i$, is simply equal to the mole fraction $Y_i$. Equation 3.17 then becomes:

$$\mathbf{d}_i = \nabla Y_i \tag{3.18}$$

and equation 3.15 simplifies to:

$$\mathbf{J}_i = \frac{C^2}{\rho} \sum_{j=1, j\neq i}^{n} M_i M_j D_{ij} \nabla Y_j - D_i^T \frac{\nabla T}{T} \tag{3.19}$$

The following identities are useful for converting between mass and mole fractions:

$$Y_i = \frac{M}{M_i} X_i \tag{3.20}$$

$$Y_i = \frac{X_i}{M_i} \left[ \sum_{j=1}^{n} \frac{X_j}{M_j} \right]^{-1} \tag{3.21}$$

where M, the average molecular weight of the mixture, can be expressed as follows:

$$M = \frac{\rho}{C} = \left[ \sum_{j=1}^{n} \frac{X_j}{M_j} \right]^{-1} \tag{3.22}$$

Using equation 3.20 and 3.22 in equation 3.19 yields

$$\mathbf{J_i} = \rho \frac{M_i}{M} \sum_{j=1, j\neq i}^{n} D_{ij} \left( \nabla X_j + X_j \frac{\nabla M}{M} \right) - D_i^T \frac{\nabla T}{T} \tag{3.23}$$

Equation 3.23 is the expression for multicomponent mass flux, $\mathbf{J}_i$, used by FLUENT in the species conservation equation.

Note from equation 3.23 that the mass flux of species $i$ depends on the concentration gradient of all of the species in the mixture. Also, as explained later in Section 3.4.2.6, there are n(n-1) independent multicomponent diffusion coefficients, $D_{ij}$, required for a mixture with n species. When n equals 10, for example, 90 values of $D_{ij}$ are required.

### 3.1.4.3 Dilute Approximation

In many CVD processes, the reacting species are present in dilute concentrations in a single carrier gas. When all species except the carrier gas are dilute, equations 3.19 and 3.23 can be greatly simplified. Appendix A develops this simplification, showing that, if the carrier gas is designated as the $n^{\text{th}}$ species, then

$$\mathbf{J}_i (i \neq n) \cong -\rho \mathcal{D}_{in} \nabla X_i - D_i^T \frac{\nabla T}{T} \tag{3.24}$$

where $\mathcal{D}_{in}$ is the binary diffusion coefficient for species $i$ in the carrier gas $n$.

The first term on the right hand side is equivalent to Fick's law of mass diffusion for a binary mixture. Thus, each dilute species diffuses approximately as if it were alone in the carrier gas, uninfluenced by the other dilute species.

Only the dilute approximation to $\mathbf{J}_i$ without the Soret diffusion term was available for simulation of species transport in FLUENT/CVD at the start of Phase II. During Phase II, the full multicomponent expression for $\mathbf{J}_i$ including the Soret term was implemented.

The dilute approximation to $\mathbf{J}_i$, equation 3.24, is used as default. The user has the option of choosing equation 3.23, however, but at the penalty of increased computational effort. For example, equation 3.24 indicates that when the dilute approximation is used, only the concentration gradient of species $i$ need be evaluated to determine the mass flux of species $i$. In contrast, all concentration gradients are required in equation 3.23. Further, only binary diffusion coefficients, $\mathcal{D}_{ij}$, are required for dilute mixtures as opposed to multicomponent diffusion coefficients, $D_{ij}$. As described in Section 3.2.4.6, $D_{ij}$ depends on concentration of the species and therefore must be calculated at all locations in the flow domain whereas the values of $\mathcal{D}_{ij}$ are constant. Lastly, only n-1 values of $\mathcal{D}_{in}$ are required for dilute mixtures as opposed to n(n-1) values of $D_{ij}$ for general mixtures.

### 3.1.5 Conservation of Energy

The expressions for conservation of energy in the fluid mixture are described in this section. The development provided by Merk [4] is followed.

Application of the law of conservation of energy to an infinitesimal control volume leads to the following expression:

$$\rho\frac{D}{Dt}(e+\frac{1}{2}\mathbf{V}\cdot\mathbf{V}) = -\nabla\cdot\mathbf{q} - \nabla\cdot\mathbf{H} - \nabla\cdot(p\mathbf{V}) + \sum_{i=1}^{n}\rho\mathbf{V_i}\cdot\mathbf{F_i} + \nabla\cdot(\tau\cdot\mathbf{V}) + Q''' \tag{3.25}$$

where: e = internal energy per unit volume
q = heat conduction vector
H = net enthalpy flux due to mass diffusion

$$\mathbf{H} = \sum_{i=1}^{n} h_i\mathbf{J}_i \tag{3.26}$$

and D/Dt is the substantive derivative defined by

$$\frac{D}{Dt} = u\frac{\partial}{\partial x} + v\frac{\partial}{\partial y} + w\frac{\partial}{\partial z} + \frac{\partial}{\partial t} \tag{3.27}$$

The right-hand side of equation 3.25 represents the various ways in which energy can be transferred or produced in the control volume. The left-hand side represents what happens to this energy—it is either stored (the transient term) or removed from the control volume by convection. The individual terms on the right-hand side represent the following effects.

$-\nabla \cdot \mathbf{q}$ = net rate of heat conduction to the control volume

$-\nabla \cdot \mathbf{H}$ = net rate of enthalpy arriving at the control volume due species diffusion

$\nabla \cdot (p\mathbf{V})$ = work done on the control volume by pressure

$\sum_{i=1}^{n} \rho \mathbf{V_i} \cdot \mathbf{F_i}$ = net work done on the control volume by external forces

$\nabla \cdot (\tau \cdot \mathbf{V})$ = net work done on the control volume by viscous stresses

$Q'''$ = volumetric heating

It is customary to eliminate the kinetic energy ($\frac{1}{2}\rho \mathbf{V} \cdot \mathbf{V}$) from equation 3.25 by forming the dot product of $\mathbf{V}$ with the momentum equation 3.4 and subtracting the result from equation 3.25:

$$\rho \frac{De}{Dt} = -\nabla \cdot \mathbf{q} - \nabla \cdot \mathbf{H} - p\nabla \cdot \mathbf{V} + \sum_{i=1}^{n} \rho \mathbf{V_i} \cdot \mathbf{F_i} + \Phi + Q''' \tag{3.28}$$

where:

$$\Phi = \tau : \nabla \mathbf{V} \tag{3.29}$$

$\Phi$ is often called the "viscous dissipation". Next, the energy equation is expressed in terms of enthalpy. This is done by using equation 3.3 to express $-p\nabla \cdot \mathbf{V}$ as follows:

$$-p\nabla \cdot \mathbf{V} = \frac{p}{\rho} \frac{D\rho}{Dt} \tag{3.30}$$

Then, the definition of enthalpy, h=e+p/$\rho$, and equation 3.30 are substituted into equation 3.28 to obtain the general form of the energy equation with enthalpy as the dependent variable:

$$\rho \frac{Dh}{Dt} = -\nabla \cdot \mathbf{q} - \nabla \cdot \mathbf{H} + \frac{Dp}{Dt} + \sum_{i=1}^{n} \rho \mathbf{V_i} \cdot \mathbf{F_i} + \Phi + Q''' \tag{3.31}$$

In the majority of CVD applications, the last four terms of equation 3.31 have only a small contribution. In this case, the energy equation simplifies to:

$$\rho \frac{Dh}{Dt} = -\nabla \cdot \mathbf{q} - \nabla \cdot \left( \sum_{i=1}^{n} h_i \mathbf{J}_i \right) \tag{3.32}$$

Equation 3.32 is the form of the energy equation implemented in FLUENT. The last term, the contribution of the species fluxes, was implemented as part of the Phase II project. (Additionally, the pressure work term, $Dp/Dt$ is available in FLUENT V3.03 and FLUENT V4.2 when required to simulate high Mach number flows.)

#### 3.1.5.1 Enthalpy-Temperature Relationship

FLUENT, as well as many other CFD codes, solve the energy equation with the mixture enthalpy as the independent variable because of computational advantages. Most boundary conditions and initial conditions, however, are specified in terms of temperatures and final

answers are most conveniently interpreted in terms of temperatures. Further, temperature is required for property evaluation. FLUENT must therefore be able to relate mixture enthalpy to temperature and vice-versa.

The specific enthalpy of the mixture is given by the following summation:

$$h = \sum_{i=1}^{n} X_i h_i \tag{3.33}$$

Since we assume that the mixture is composed of ideal gasses, $h_i$ is a function of temperature only. Thus,

$$dh_i = C_{p,i} dT \tag{3.34}$$

which may be integrated to obtain $h_i$ used in equation 3.33:

$$h_i = h^{\circ}_{f,i} + \int_{T_o}^{T} C_{p,i} dT \tag{3.35}$$

$h^{\circ}_{f,i}$ is the specific enthalpy of formation of component i from its constitutive elements at a reference temperature $T_o$. Substituting equation 3.35 into equation 3.33 yields the expression for mixture enthalpy in terms of temperature.

$$h = \sum_{i=1}^{n} X_i \left( h^{\circ}_{f,i} + \int_{T_o}^{T} C_{p,i} dT \right) \tag{3.36}$$

Obtaining the temperature as a function of mixture enthalpy requires inverting the functional dependence between $h$ and $T$ in equation 3.36. This may be done explicitly when all $C_{p,i}$ are constant or linear. In the more general case, $T$ is found from $h$ by iterative means.

### 3.1.5.2 Heat of Reaction

The energy liberated by chemical reactions is not explicitly seen in equation 3.31. Rather, as shown below, the heats of reaction are included in $Dh/Dt$. Taking the substantive derivative of both sides of equation 3.33 yields:

$$\frac{Dh}{Dt} = \sum_{i=1}^{n} h_i \frac{DX_i}{Dt} + \sum_{i=1}^{n} X_i \frac{Dh_i}{Dt} \tag{3.37}$$

Multiplying both sides by $\rho$, and substituting equations 3.10 and 3.34, equation 3.37 becomes:

$$\rho \frac{Dh}{Dt} = \sum_{i=1}^{n} h_i (-\nabla \cdot \mathbf{J_i}) + \sum_{i=1}^{n} h_i R_i + \rho \sum_{i=1}^{n} X_i C_{p,i} \frac{DT}{Dt} \tag{3.38}$$

The second term on the right hand side of equation (3.38) is rearranged using equation 3.12:

$$\sum_{i=1}^{n} h_i R_i = \sum_{i=1}^{n} h_i \sum_{k=1}^{K} R_{ik} = \sum_{k=1}^{K} \sum_{i=1}^{n} h_i R_{ik} \tag{3.39}$$

By definition, the heat of reaction for reaction $k$, $Q_{R,k}$, is related to $R_{ik}$ as follows:

$$\frac{dQ_{R,k}}{dt} = -\sum_{i=1}^{n} h_i R_{ik} \tag{3.40}$$

provided that the enthalpies $h_i$ are related to a common reference, such as equation 3.35. Then, substituting equations 3.39 and 3.40 into equation 3.38:

$$\rho\frac{Dh}{Dt} = -\sum_{i=1}^{n} h_i(\nabla \cdot \mathbf{J_i}) - \sum_{k=1}^{K} \frac{dQ_{R,k}}{dT} + \rho\frac{DT}{Dt}\sum_{i=1}^{n} X_i C_{p,i} \tag{3.41}$$

Equation 3.41 shows that energy liberated or absorbed by virtue of chemical reaction (second term on the right hand side) is automatically accounted for by the form of energy equation used, i.e., equation 3.31, as long as all species enthalpies are evaluated with respect to a common reference temperature.

#### 3.1.5.3 The Heat Flux Vector

The heat flux vector, $\mathbf{q}$, is comprised of one component due to gradients in temperature (Fourier's law of heat conduction) and another component due to gradients in chemical species (the Dufour effect) [4]. The present implementation neglects the Dufour effect. The heat flux vector then becomes:

$$\mathbf{q} = -k\nabla T \tag{3.42}$$

## 3.2 CHEMICAL REACTIONS

Prior to Phase I, chemical reaction modeling in FLUENT was limited to only one reaction with two reactants and one product. This was suitable for simulating the global aspects of some combustion problems but not for CVD. During Phase I, the chemical reaction modeling capabilities described in this section were implemented in FLUENT/CVD. During Phase II, these capabilities remained in the Phase II version of FLUENT/CVD and have been implemented into the released codes FLUENT V3.03 and FLUENT V4.2.

### 3.2.1 Reaction Stoichiometry

FLUENT permits multiple simultaneous reactions. The reactions can occur on the surface and/or in the gas. Any species in the gas mixture may participate in each reaction as a reactant and/or a product. The stoichiometry of a chemical reaction is represented as shown in the following reaction equation:

$$\sum_{i=1}^{n} \nu'_{ik} \mathcal{M}_i \longrightarrow \sum_{i=1}^{n} \nu''_{ik} \mathcal{M}_i \tag{3.43}$$

where: $\nu'_{ik}$ = stoichiometric coefficient of species $i$ as a reactant in reaction $k$

$\nu''_{ik}$ = stoichiometric coefficient of species $i$ as a product in reaction $k$

$\mathcal{M}_i$ = chemical symbol for species $i$

n = total number of chemical species

Each reaction is defined by specifying the stoichiometric coefficients of the reaction. For example, consider the following reaction mechanism for silicon deposition from silane:

$$\begin{aligned}
Reaction\ 1\ (surface): &\quad SiH_4(g) \longrightarrow Si(s) + 2H_2(g) \\
Reaction\ 2\ (gas): &\quad SiH_4(g) \longrightarrow SiH_2(g) + H_2(g) \\
Reaction\ 3\ (gas): &\quad SiH_2(g) \longrightarrow Si(g) + H_2(g) \\
Reaction\ 4\ (surface): &\quad SiH_2(g) \longrightarrow Si(s) + H_2(g) \\
Reaction\ 5\ (surface): &\quad Si(g) \longrightarrow Si(s)
\end{aligned} \tag{3.44}$$

There are a total of five chemical species in this reaction mechanism:

$$SiH_4(g) \qquad SiH_2(g) \qquad H_2(g) \qquad Si(s) \qquad Si(g)$$

Note that a chemical species on the surface is treated as distinct from the same chemical species in the gas. Thus, $Si(s)$ and $Si(g)$ are two distinct species. The stoichiometric coefficients corresponding to the reactions in equation 3.44 are shown in Table 3.3.

Table 3.3: **Stoichiometry Coefficients for the Reaction Mechanism of Equation 3.44**

| | | CHEMICAL SPECIES | | | | |
|---|---|---|---|---|---|---|
| | | $SiH_4$ | $SiH_2(g)$ | $H_2(g)$ | $Si(s)$ | $Si(g)$ |
| Reaction | 1 | 1 | 0 | -2 | -1 | 0 |
| | 2 | 1 | -1 | -1 | 0 | 0 |
| | 3 | 0 | 1 | -1 | 0 | -1 |
| | 4 | 0 | 1 | -1 | -1 | 0 |
| | 5 | 0 | 0 | 0 | -1 | 1 |
| $\nu'_{ik}$ (reactant) indicated by positive value<br>$\nu''_{ik}$ (product) indicated by negative value | | | | | | |

### 3.2.2 Forward/Reverse Reactions

The forward and reverse directions of a particular chemical reaction equation are considered as two distinct chemical reactions. The stoichiometric coefficients of a reaction in the reverse direction are opposite in sign to those of the corresponding forward direction reaction.

### 3.2.3 Gas-Phase and Surface Reactions

Each chemical reaction is declared by the user to occur in either the gas-phase or on surfaces adjacent to the fluid domain. Reactions 1, 4, and 5 of the reaction mechanism of equation 3.44, for example, occur only on surfaces; reactions 2 and 3 occur only in the gas. Identical reactions are allowed both in the gas phase and on the surface, but are treated as two distinct reactions (c.f., reactions 3 and 4 of equation 3.44). Surface reactions may be allowed to occur only on portions of the solid boundaries of the flow domain while the other solid boundaries remain free of surface reactions.

Surface reactions are treated differently from gas-phase reactions in the following two ways:

1. For a gas-phase reaction, the rate of creation or destruction of chemical species is used in the source term, $R_i$, of the species conservation equation 3.10. For a surface reaction, the reaction rate is used in the boundary condition for the species transport equation, c.f., equation 3.55.

2. A gas-phase reaction rate is defined on a volumetric basis whereas the reaction rate for a surface reaction is defined for unit surface area.

### 3.2.4 Gas-Phase and Surface Species

Each species is declared by the user to be either a gas-phase or surface species. As described in Section 3.2.1, a chemical species which exists in both the gas and on surfaces is modeled as two distinct species. The distinctions between gas-phase and surface species are that species conservation equations are solved only for gas-phase species and mass deposition is allowed only for surface species.

Surface species are permitted only in surface reactions.

### 3.2.5 Reaction Rate Expressions

A reaction rate must be specified for each chemical reaction. There are two ways to define the reaction rate. An Arrhenius-type expression has been incorporated as a default option. When using the default Arrhenius reaction rate, the user only needs to provide suitable constants in the Arrhenius expression. An Arrhenius-type expression is not always a suitable model, however, for the reaction rate. In such cases, the user has the option to define a unique

expression for the reaction rate via a "user-defined subroutine". That is, the user writes a Fortran language subroutine which implements the desired expression for the reaction rate. This subroutine is then compiled and linked with the object-code version of FLUENT.

#### 3.2.5.1 Arrhenius Rate Expression

FLUENT uses an Arrhenius-type expression as the default expression for calculating chemical reaction rates. The default expression can be written in several forms which differ primarily in the variable used for the concentration of the reacting species: mass fraction mole concentration, or partial pressure. Different but numerically equivalent forms of the rate expressions are listed below.

$$R_{jk} = (\nu''_{jk} - \nu'_{jk}) M_j T^{\beta_k} \left[ A_k e^{-E_k/RT} \prod_{\text{react}} C_j^{\nu_{jk}} \right] \tag{3.45a}$$

$$\mathcal{R}_{jk} = (\nu''_{jk} - \nu'_{jk}) T^{\beta_k} \left[ A_k e^{-E_k/RT} \prod_{\text{react}} C_j^{\nu_{jk}} \right] \tag{3.45b}$$

$$R_{jk} = (\nu''_{jk} - \nu'_{jk}) \rho^{\xi_k} Z_{jk} T^{\beta_k} M_j \left[ A_k e^{-E_k/RT} \prod_{\text{react}} X_j^{\nu_{jk}} \right] \tag{3.45c}$$

$$\mathcal{R}_{jk} = (\nu''_{jk} - \nu'_{jk}) \frac{T^{\beta_k}}{(RT)^{\xi_k}} \left[ A_k e^{-E_k/RT} \prod_{\text{react}} P_j^{\nu_{jk}} \right] \tag{3.45d}$$

where: $R_{jk}$ = mass rate of production of species $j$ by reaction $k$, kg/m$^3$-s for a gas phase reaction and kg/m$^2$-s for a surface reaction.

$\mathcal{R}_{jk}$ = mole rate of production of species $j$ by reaction $k$, kg-mole/m$^3$-s for a homogeneous reaction and kg-mole/m$^2$-s for a surface reaction.

$X_j$ = mass fraction of species $j$, no units

$C_j$ = mole concentration of species $j$, kg-mole/m$^3$

$P_j$ = partial pressure of species $j$, Pa

$\rho$ = local mixture density, kg/m$^3$

T = local temperature, K

$\underline{A_k}$ = pre-exponential factor, units consistent with other terms

$\underline{E_k}$ = activation energy, J/kg-mole

$\underline{\beta_k}$ = temperature exponent of reaction rate expression, no units

R = universal gas constant, 8314 J/kg-mole-K

$\underline{\nu_{jk}}$ = concentration exponent of species $j$ in reaction $k$
of the reaction rate expression, reactants only, no units

$\underline{\nu'_{jk}}$ = stoichiometric coefficient of species $j$ as a reactant in reaction $k$, no units

$\underline{\nu''_{jk}}$ = stoichiometric coefficient of species $j$ as a product in reaction $k$, no units

$\underline{M_j}$ = molecular weight of species $j$, kg/kg-mole

$\xi_k = \sum \nu_{jk}$(reactants only)

$Z_{jk} = (\prod_{\text{reactants}} M_j^{\nu_{jk}})^{-1}$

The rate expression based on mass fraction X, equation 3.45c, is used internally in FLUENT. The other expressions are useful when evaluating rate constants from data expressed in alternative forms.

The user input parameters are underlined in the list of variables which follow equation 3.45. The user inputs which define the Arrhenius reaction rate do not depend on which form of equation 3.45 is used. The units of $A_k$, however, depend on the numerical values of $\nu_{jk}$ and $\beta_k$ and must be consistent with the convention of volumetric rates for gas-phase reactions and areal rates for surface reactions.

#### 3.2.5.2 User-Defined Reaction Rate

In addition to the default Arrhenius rate expression, FLUENT allows the user to provide arbitrary reaction rate expressions via user-supplied subroutines. The user implements the desired rate expression in Fortran language subroutines. The subroutine is then compiled to create an object code version and linked to the object version of FLUENT.

The values of pressure, temperature and mass fraction of any of the chemical species are available to the user subroutine. Thus, the user-defined reaction rate may be expressed as functions of any of these variables.

In order to enhance stability of convergence, each user-specified reaction rate expression must be linearized according to the following expression:

$$R_{jk} = A_{jk} + B_{jk} X_j \tag{3.46}$$

It is the values of the parameters $A_{jk}$ and $B_{jk}$ in equation 3.46 which are calculated in the user-defined subroutine. These values are then passed as arguments back to FLUENT

for calculation of the rate according to equation 3.46. The user-defined subroutine must therefore include expressions for $A_{jk}$ and $B_{jk}$ for each chemical species in each reaction for which the user wishes to define a reaction rate expression.

$A_{jk}$ and $B_{jk}$ may be functions of pressure, temperature and mass fraction of any of the chemical species in the mixture. Since there are two expressions (one for $A_{jk}$ and one for $B_{jk}$) required to define a single reaction rate, there is some latitude in choosing the particular expressions for $A_{jk}$ and $B_{jk}$. In order to ensure convergence of solution, however, it is required that the value $B_{jk}$ be less than or equal to zero [7]. An expression for $B_{jk}$ which satisfies this requirement can always be developed by suitable adjustment of the expressions for both $A_{jk}$ and $B_{jk}$.

As an example of the development of the expressions for $A_{jk}$ and $B_{jk}$, we use the expression of Jensen and Graves [8] for the surface reaction rate of silicon from silane for a one-step, global reaction mechanism:

$$SiH_4 \longrightarrow Si(s) + 2H_2 \tag{3.47}$$

$$R_{Si} = \frac{kP_{SiH_4}}{1 + K_1 P_{H_2} + K_2 P_{SiH_4}} \tag{3.48}$$

where: $k, K_1, K_2$ = are constants
$P_{SiH_4}$ = partial pressure of silane
$P_{H_2}$ = partial pressure of hydrogen
$R_{Si}$ = deposition rate of silicon

Converting the partial pressures of equation 3.48 to mass fractions, and using the stoichiometry of equation 3.47 to convert rates of silicon to rates of silane and hydrogen, the values of $A_{jk}$ and $B_{jk}$ for silane become:

$$A = 0 \tag{3.49a}$$

$$B = -K_3 \left(\frac{M_{SiH_4}}{M_{Si}}\right) \left[\frac{k\Upsilon_{SiH_4}}{1 + K_1 \Upsilon_{H_2} Y_{H_2} + K_2 \Upsilon_{SiH_4} Y_{SiH_4}}\right] \tag{3.49b}$$

and for $H_2$:

$$A = K_3 \left(\frac{2M_{H_2}}{M_{Si}}\right) \left[\frac{kY_{SiH_4}\Upsilon_{SiH_4}}{1 + K_1 \Upsilon_{H_2} Y_{H_2} + K_2 \Upsilon_{SiH_4} Y_{SiH_4}}\right] \tag{3.50a}$$

$$B = 0 \tag{3.50b}$$

where: $\Upsilon_{SiH_4} = \rho TR/M_{SiH_4}$
$\Upsilon_{H_2} = \rho TR/M_{H_2}$
$K_3 =$ conversion constant

The source code which implements equations 3.49a and 3.50a is provided in Appendix B.

## 3.3 BOUNDARY CONDITIONS

Solution of the conservation equations described in Section 3.1 requires that boundary conditions for the solution variables–fluid velocities, temperature and species concentrations–be specified on the boundaries of the solution domain. Boundaries types of particular relevance for CVD simulations are:

1. inlet
2. outlet
3. fluid-wall interface

The boundary conditions typically used for CVD simulation are described in the following section for each of these boundary types.

FLUENT also implements other boundary types (e.g. conducting wall, symmetry, porous). These are often useful for CVD simulation but aren't directly related to the work done in this project. These boundary types are described in the standard FLUENT documentation (e.g., [3]).

### 3.3.1 Inlet

An inlet boundary is used to introduce the fluid mixture into the solution domain. The values of all solution parameters for the inlet fluid must be provided by the user:

- velocity components in the coordinate directions or gas pressure
- temperature
- mass fraction of each chemical species

With some restrictions, these variables may be functions of position at the inlet.

Except for increasing the number of chemical species, the inlet boundary conditions were not altered during the SBIR project.

### 3.3.2 Outlet

The fluid mixture leaves the solution domain at outlet boundaries. A precise specification of the solution parameters at an outlet is impossible since the flow conditions at the outlet of the physical hardware depend on the nature of the hardware downstream of the outlet and this downstream hardware is not part of the simulation. As is typical of most CFD simulation algorithms, FLUENT applies a gradient boundary condition at outlets; namely, the gradient of all solution variables is taken to be zero.

Since the outlet boundary condition is not theoretically correct, it is important to properly choose the location of the outlet in the solution domain. The outlet should be a region of small gradients in the solution variables and/or located sufficiently downstream from regions of principal interest in the solution domain.

### 3.3.3 Fluid-Wall

The interface between the fluid and solid boundaries requires boundary conditions for the solution parameters.

#### 3.3.3.1 Fluid Velocity

FLUENT provides two boundary conditions for velocity at fluid-wall boundaries. The default boundary condition is zero velocity, i.e., "no-slip". Also available is the "slip" or "no shear" boundary condition where the gradients of the velocity components parallel to the wall are zero and the value of velocity normal to the wall is zero.

#### 3.3.3.2 Temperature

FLUENT provides three temperature boundary conditions at fluid-wall boundaries commonly encountered for CVD simulation. The first, and simplest, is specification of the wall temperature at the solid boundary; the fluid temperature adjacent to the wall takes the value of the wall temperature. The second is the heat flux boundary condition whose mathematical description is shown below:

$$q'' = -k\frac{\partial T}{\partial n} \tag{3.51}$$

where: $q''$ = surface heat flux into the fluid
$k$ = thermal conductivity of the fluid adjacent to the surface
$\partial T/\partial n$ = gradient of temperature along the normal to the surface in the direction toward the fluid

For these boundary conditions, the user specifies the value of either the surface temperature or heat flux.

The third boundary condition comes into effect when the adjoining wall is a thermally conducting region. In this case, the temperature of the adjoining wall region is a calculated solution parameter and neither the interface temperature or heat flux is known *a priori*. Under these conditions, the boundary condition relates the balance of heat fluxes at the interfaces as follows:

$$-\left(k\frac{\partial T}{\partial n}\right)_w = -\left(k\frac{\partial T}{\partial n}\right)_f + \sum_{k=1}^{K} q''_{k,surface} \tag{3.52}$$

where the nomenclature of the gradient terms is the same as for equation 3.51 except that subscript "w" designates within the wall at the interface and subscript "f" designates within the fluid at the interface. The last term in equation 3.52 represents the heat liberated by all surface reactions occurring at the interface. The value of the heat of reaction, $q''_k$, is calculated as follows:

$$q''_k = -\sum_{i=1}^{n} h_i R_{ik} \tag{3.53}$$

where reaction $k$ is a surface reaction, $R_{ik}$ is the reaction rate per unit area, and $h_i$ is calculated at the local surface temperature according to equation 3.35.

The heat of the reaction term of equation 3.52 was implemented as part of the Phase II project.

#### 3.3.3.3 Chemical Species

The general boundary condition for chemical species at a surface expresses the conservation of each of the species at the interface:

$$-\mathbf{J}_i \cdot \mathbf{n} + R_{i,surface} - \rho X_i(\mathbf{V} \cdot \mathbf{n}) = 0 \tag{3.54}$$

The first term in equation 3.54 represents the arrival at the interface of species $i$ by diffusion, either ordinary mass diffusion or thermal diffusion. $\mathbf{J_i}$ is defined in equation 3.11 and $\mathbf{n}$ is the normal to the surface pointing toward the fluid. The second term in equation 3.54 is the rate of creation of species $i$ due to all surface reactions. The last term in equation 3.54 represents the arrival of species $i$ at the surface due to bulk convection at the surface.

The convective (third) term of equation 3.54 comes into effect only when there are deposition (or etching) reactions on the surface. When there are no such reactions, the velocity component normal to the solid interface is identically zero. When there are deposition (or etching) reactions, however, there is a net mass flux to (or from) the surface which indicates that the velocity normal to the surface is not zero. For typical CVD applications, this velocity is very small compared to the velocity in the bulk and it can be neglected in the momentum and energy boundary conditions. For some conditions with non-dilute mixtures, however, its impact on the species boundary condition, equation 3.54, can become noticeable.

The convective term in equation 3.54 is not implemented in FLUENT V3.03. It will be implemented in FLUENT V4.2.

For surfaces without deposition (or etching) reactions, equation 3.54 becomes:

$$-\mathbf{J}_i \cdot \mathbf{n} + R_{i,surface} = 0 \tag{3.55}$$

For illustration, $\mathbf{J}_i$ is expressed using the dilute approximation, i.e., equation 3.24:

$$\rho \mathcal{D}_{in} \frac{\partial X_i}{\partial n} + D_i^T \frac{\partial T}{\partial n} + R_{i,surface} = 0 \tag{3.56}$$

If there are no surface reactions on the wall, then $R_{i,surface} = 0$ and equation 3.56 becomes:

$$\rho \mathcal{D}_{in} \frac{\partial X_i}{\partial n} + D_i^T \frac{1}{T} \frac{\partial T}{\partial n} = 0 \tag{3.57}$$

Thus, for non-reacting walls, the species boundary condition becomes a balance between mass diffusion by virtue of the driving forces of concentration and temperature gradients. Finally, if thermal diffusion can be neglected, equation 3.57 becomes:

$$\frac{\partial X_i}{\partial n} = 0 \tag{3.58}$$

Equation 3.58 is the simplest species wall boundary condition implemented by FLUENT.

At the start of this SBIR project, only equation 3.58 was available for the species boundary condition. The other effects were introduced during Phase I and Phase II of the SBIR program.

## 3.4 THERMOPHYSICAL PROPERTIES

Solution of the complete set of transport equations described in Section 3.1 requires values for the thermophysical properties listed in Table 3.4. The values of these properties, in general, change from one location in the flow field to another depending on the local values of pressure, temperature and chemical composition.

Prior to the Phase II project, all properties except density were not composition dependent and only binary mass diffusivities, as opposed to multicomponent mass diffusivities, were available for modeling species transport. For CVD simulation, therefore, only mixtures with dilute concentrations of reacting species in a carrier gas could be properly modeled. One of the major efforts in Phase II was to enhance property evaluation capabilities consistent with the enhancements made to the transport equations. As a result, all transport properties may now be temperature and composition dependent and multicomponent diffusion coefficients and thermal diffusion coefficients have been implemented for the first time. Table 3.5 shows the property modeling enhancements made during the Phase II work.

Table 3.4: **Thermophysical Properties Needed for Solution of the Transport Equations**

| Symbol | Property |
|---|---|
| $\rho$ | Density |
| $\mu$ | Viscosity |
| $k$ | Thermal Conductivity |
| $Cp$ | Specific Heat at Constant Pressure |
| $h_f^o$ | Enthalpy of Formation |
| $\mathcal{D}_{ij}$ | Binary Diffusion Coefficient |
| $D_{ij}$ | Multicomponent Diffusion Coefficient |
| $D_i^T$ | Soret Diffusion Coefficient |

Not all of the properties listed in Table 3.4 are required for every simulation. The properties required for a particular simulation depend on the modeling strategy employed. In particular, most CVD processes involve all of the transport phenomena discussed in Section 3.1. Under the most comprehensive modeling strategy, values for all of the transport properties listed in Table 3.4 are therefore required. If, however, the Soret effect can be neglected, the thermal diffusion coefficient $D_i^T$ is not required. Or, if the gas mixture can be approximated as a dilute mixture, only the binary diffusion coefficients $\mathcal{D}_{ij}$ are required as opposed to the more numerous multicomponent diffusion coefficients $D_{ij}$.

A considerable degree of flexibility has been incorporated for evaluating physical properties. Modeling options are available at each of the following levels:

**Property Models:** The fluid may be modeled (1) as an ideal gas or not an ideal gas, and (2) with mixture properties composition dependent or not composition dependent.

**Property Expressions:** For any property modeling option, there is at least one and usually more than one expression available to evaluate physical properties. In addition, a user-supplied Fortran subroutine may be used for evaluation of composition dependent mixture properties if the built-in expressions are not appropriate.

The choice of property models is discussed in Section 3.4.1. The expressions implemented for evaluation of properties are described in Section 3.4.2

Table 3.5: **Transport Property Modeling Enhancements**

| | Before Phase II | | | Phase II Capabilities | | |
|---|---|---|---|---|---|---|
| **Property** | **Temperature Dependent** | **Composition Dependent** | **Pressure Dependent** | **Temperature Dependent** | **Composition Dependent** | **Pressure Dependent** |
| $\rho$ | • | •(1) | • | • | • | • |
| $\mu$ | • | | | • | • | |
| k | • | | | • | • | |
| $C_p$ | • | | (2)(3) | • | • | |
| $\mathcal{D}_{ij}$ | • | | • | • | • | • |
| $D_{ij}$ | | | | • | • | • |
| $D_i^T$ | | | | • | • | |

(1) four species maximum
(2) temperature dependence of $\mu/\rho$
(3) pressure dependence of $\rho$

## 3.4.1 Property Models

The fluid type is specified by the user by selecting from each of the options below:

1. Ideal gas vs. not ideal gas.
2. Composition dependent vs. composition independent properties.

The mathematical expressions used for evaluating properties, described in Section 3.4.2, depend on the property model chosen.

### 3.4.1.1 Ideal Gas Vs. Not Ideal Gas

The choice of the ideal gas property model (1) enables fluid density to be calculated by the ideal gas law and, (2) if desired, enables other transport properties to be calculated by built-in expressions based on kinetic theory. The built-in kinetic theory property expressions are not available if the fluid is not modeled as an ideal gas. The ideal gas approximation is appropriate for most CVD applications but, for example, is not suitable for liquids.

### 3.4.1.2 Composition Dependent Vs. Composition Independent Properties

The user has the option, with certain restrictions described below, to model properties as being composition independent or composition dependent. Composition independent properties may be suitable, for example, when reactants are present in dilute concentrations in the carrier gas. In this case, the values of $C_p$, $\mu$, and $k$ are associated with the carrier gas and the values of $\mathcal{D}_{ij}$ are considered to be those of each species diffusing as if it were alone in the carrier gas. The option of composition dependent properties is more suitable in

cases where the mixture of species is not dilute, especially when the composition is spatially varying due to mass diffusion and chemical reaction.

The option for composition dependency may be selected separately for each transport property.

### 3.4.2 Property Expressions

Transport properties for the pure species are computed during the course of the solution based on local values of temperature and pressure. The functional relationships between pure species properties and temperature and pressure are provided by expressions which are already incorporated into the software. There is a choice of expressions for each property except enthalpy of formation; the choice is made by menu selection. Thus, for example, the pure component viscosity of one of the species in the mixture may be described as a polynominal function of temperature or in terms of an expression derived from kinetic theory. The user must provide appropriate numerical constants for each default expression chosen; thus, for the pure component viscosity example cited above, the numerical constants would be coefficients of each term in the polynomial function of temperature or values of Lennard-Jones parameters for the kinetic theory expression.

Transport properties of a mixture of species are calculated based on the values of the pure species properties and the local composition. Default expressions for most of the mixture properties are already incorporated into the software and are accessed by menu selection. These are called "menu-accessed" default expressions. Additionally, mixture properties can be calculated from expressions supplied in user-defined Fortran subroutines. Several user-defined subroutines have expressions already programmed which are different from the menu-accessed default expression; these are called the "user-subroutine" default expressions. The user-supplied subroutines can be reprogrammed by the user, providing added flexibility for calculation of mixture properties.

Sections 3.4.2.1 through 3.4.2.8 describe the menu-accessed and user-subroutine default expressions and required user inputs for the properties listed in Table 3.4. For reference, Appendix C lists the available default expressions. Section 3.4.2.9 describes the use of user-defined subroutines for mixture property calculation when an expression different from the user-subroutine default expression is desired.

#### 3.4.2.1 Fluid Density

**3.4.2.1.1 Pure Species** When the ideal gas model is enabled, the density of a pure gas is computed as follows:

$$\rho = \frac{P\mathcal{M}}{RT} \tag{3.59}$$

where $\mathcal{M}$ is the molecular weight of the gas, $R$ is the universal gas constant and $P$ and $T$ are the local values of pressure and temperature, respectively. The molecular weight is a user input while $P$ and $T$ are a result of the solution.

When the ideal gas model is disabled, the pure-species density is expressed as either a polynomial or piecewise-linear function of temperature. For example, a polynomial function of fourth order is:

$$\rho(T) = a_0 + a_1 T + a_2 T^2 + a_3 T^3 + a_4 T^4 \tag{3.60}$$

The coefficients $a_0$ through $a_4$ are menu-prompted user inputs. A piecewise-linear function is shown graphically in Figure 3.1. The end-point values ($T$ and $P$ in Figure 3.1) are menu-prompted user inputs.

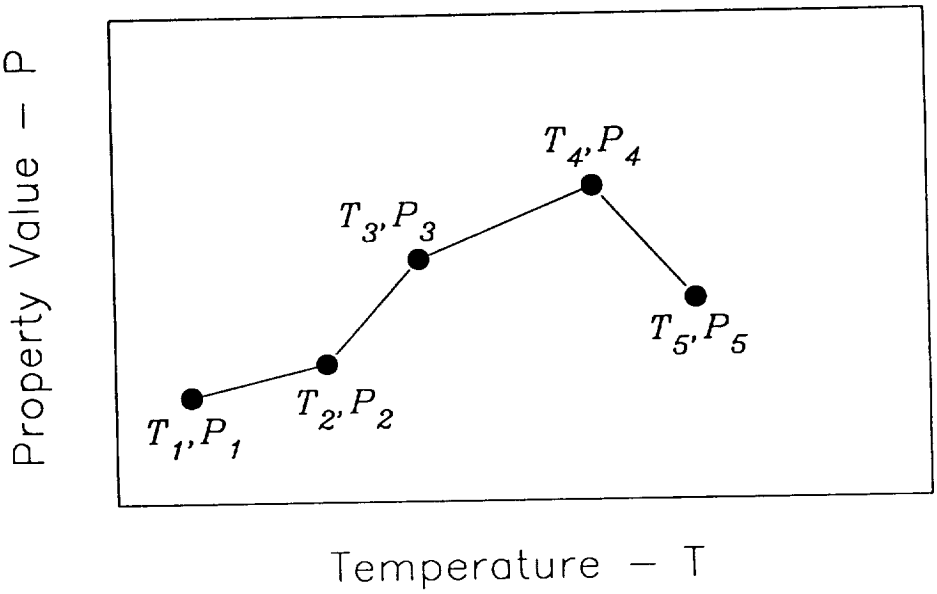


Figure 3.1: Graphical Representation of a Piecewise-Linear Function

**3.4.2.1.2 Composition Dependent** The density of an ideal gas mixture depends on the mixture composition as follows:

$$\rho_{\text{mix}} = \frac{P}{RT \sum_{i=1}^{n} \frac{X_i}{\mathcal{M}_i}} \tag{3.61}$$

where $X_i$ is the mass fraction of species $i$ in the mixture, $\mathcal{M}_i$ is the molecular weight of species $i$, and $n$ is the number of species in the fluid mixture. The values of $P$, $T$, and $X_i$ in equation 3.61 are calculated during the solution; the values of $\mathcal{M}_i$ are user inputs.

When the ideal gas model is disabled, the mixture density is computed as the mass fraction weighted average of the pure species densities:

$$\rho_{\text{mix}} = \sum_{i=1}^{n} \rho_i X_i \tag{3.62}$$

where $\rho_i$ is the density of pure species $i$ and the summation is taken only over fluid phase species.

A user-defined subroutine for composition dependent density is not available.

### 3.4.2.2 Fluid Viscosity

**3.4.2.2.1 Pure Species** When the ideal gas property model is disabled, the viscosity of pure species $i$, $\mu_i$, is expressed as either a polynomial or piecewise-linear function of temperature.

When the ideal gas property model is enabled, $\mu_i$ can be expressed as a polynomial or piecewise-linear function of temperature, or from an expression derived from kinetic theory [9]:

$$\mu_i = 2.67(10^{-6}) \frac{\sqrt{\mathcal{M}_i T}}{\sigma_i^2 \Omega_\mu} \tag{3.63}$$

All quantities are in SI units except $\sigma_i$(Å). The dimensionless function $\Omega_\mu$ depends on $T^*$, here defined as:

$$T^* = \frac{T}{(\epsilon/\kappa)_i} \tag{3.64}$$

The relationship between $\Omega_\mu$ and $T^*$ is taken from [10] and implemented in table format for values of $T^*$ between 0.3 and 100. $\sigma_i$ and $(\epsilon/\kappa)_i$ are the Lennard-Jones parameters of species $i$ and are menu-prompted user inputs.

**3.4.2.2.2 Composition Dependent** Composition dependent mixture viscosity is expressed as a function of the pure component concentrations and viscosities. The user must therefore provide information for the viscosity of all pure gas phase species when modeling the mixture viscosity as composition dependent.

When the ideal gas property model is enabled, the menu-accessed default expression for mixture viscosity is calculated from a semi-empirical expression derived from kinetic theory as recommended by [9]:

$$\mu_{\text{mix}} = \sum_{i=1}^{n} \frac{Y_i \mu_i}{\sum_{j=1}^{n} Y_j \phi_{ij}} \tag{3.65}$$

$$\phi_{ij} = \frac{\left[1 + \left(\frac{\mu_i}{\mu_j}\right)^{\frac{1}{2}} \left(\frac{\mathcal{M}_j}{\mathcal{M}_i}\right)^{\frac{1}{4}}\right]^2}{\left[8\left(1 + \frac{\mathcal{M}_i}{\mathcal{M}_j}\right)\right]^{\frac{1}{2}}} \tag{3.66}$$

A simpler, empirically derived expression, called the "square-root" law, is used as the user-subroutine default expression for mixture viscosity [11]:

$$\mu_{\text{mix}} = \frac{\sum_{i=1}^{n} \sqrt{\mathcal{M}_i}\ Y_i \mu_i}{\sum_{i=1}^{n} \sqrt{\mathcal{M}_i}\ Y_i} \tag{3.67}$$

When the ideal gas property model is disabled, the menu-accessed default expression for mixture viscosity is the mass fraction average of the pure species viscosities:

$$\mu_{\text{mix}} = \sum_{i=1}^{n} X_i \mu_i \tag{3.68}$$

### 3.4.2.3 Fluid Specific Heat At Constant Pressure

**3.4.2.3.1 Pure Species** The constant pressure specific heat of a pure species $i$, $C_{p,i}$, can be expressed as either a polynomial or piecewise-linear function of temperature when the ideal gas property model is either enabled or disabled. When the ideal gas model is enabled, a kinetic theory based-expression is also available:

$$C_{p,i} = \frac{1}{2}\ \frac{R}{\mathcal{M}_i}(f_i + 2) \tag{3.69}$$

where $f_i$ is the number of modes of energy storage (degrees of freedom) for species $i$. The value of $f_i$ is a constant and is a menu-prompted user input.

The accuracy of equation 3.69 depends on the complexity of the molecule of species $i$. For monatomic gasses, $f_i = 3$ (one for each direction of translation) and the value of $C_{p,i}$ from equation 3.69, $\frac{5}{2}R/\mathcal{M}_i$, is very close to measured values. For polyatomic gases, rotational and vibrational degrees of freedom are present and $f_i$ is larger than 3. A diatomic molecule such as $H_2$, for example, can have two rotational and two vibrational degrees of freedom in addition to the three translational degrees of freedom. The vibrational degrees of freedom are activated, however, only at elevated temperatures. Thus, for diatomic molecules at typical CVD temperatures, $f_i = 5$ and $C_{p,i} = \frac{7}{2}R/\mathcal{M}_i$ provides a reasonable approximation. As the molecule becomes more complex, however, estimation of the number of degrees of freedom and the temperatures at which they are active becomes a serious impediment to the use of equation 3.69. In such cases, equation 3.69 should be used only to provide a reasonable order of magnitude value for $C_{p,i}$ when no other data is available. An alternate approach to using equation 3.69 is to use one of the empirical methods which have been developed to estimate the specific heats of complex molecules. The values of $C_{p,i}$ so obtained can then be approximated as a polynomial or piecewise-linear function of temperature.

**3.4.2.3.2 Composition Dependent** The value of mixture specific heat is calculated as the mass fraction average of the values of the pure species specific heats:

$$C_p = \sum_{i=1}^{n} X_i C_{p,i} \tag{3.70}$$

Equation 3.70 is used for all default options: ideal gas property model enabled or disabled; menu-accessed or user-subroutine default.

### 3.4.2.4 Fluid Thermal Conductivity

**3.4.2.4.1 Pure Species** When the ideal gas property model is enabled or disabled, the thermal conductivity of pure species $i$, $k_i$, may be expressed as either a polynomial or piecewise-linear function of temperature.

When the ideal gas property model is enabled, a kinetic theory based expression for $k_i$ is available. The thermal conductivity of a pure monatomic gas is generally related to $\mu_i$ by a constant. For polyatomic gases, however, the relationship between $k_i$ and $\mu_i$ is more complex due to additional degrees of freedom which also transport energy in response to a temperature gradient. Specific heat information is therefore typically incorporated to calculate $k_i$. Different formulas have been developed to incorporate the specific heat data into the calculation of $k_i$ (e.g., [12]), sometimes with dependencies on the type of molecule (e.g., organic, inorganic, polar, elongated). A relatively simple expression employing the Euken correction [10] is employed here:

$$k_i = \frac{15}{4} \frac{R}{\mathcal{M}_i} \mu_i \left[ \frac{4}{15} \frac{C_{p,i} R}{\mathcal{M}_i} + \frac{1}{3} \right] \tag{3.71}$$

For a monatomic gas, $C_{p,i}/\mathcal{M}_i = 5\mathrm{R}/2$ and the expression within brackets of equation 3.71 reduces to unity.

**3.4.2.4.2 Composition Dependent** Composition dependent mixture thermal conductivity is expressed as a function of the pure component concentrations and thermal conductivities. The user must therefore provide information to allow calculation of the thermal conductivities of all pure species when modeling the mixture thermal conductivity as composition dependent.

When the ideal gas property model is enabled, the menu-accessed default expression for mixture thermal conductivity is calculated from a semi-empirical expression derived from kinetic theory recommended by [9]:

$$k_{\mathrm{mix}} = \sum_{i=1}^{n} \frac{Y_i k_i}{\sum_{j=1}^{n} Y_j \phi_{ij}} \tag{3.72}$$

where, $\phi_{ij}$ is given in 3.66. Equation 3.72 is reported to calculated $k_{\mathrm{mix}}$ within 4% of measured values for mixtures containing non-polar polyatomic gases including $CH_4$, $O_2$, $N_2$, $C_2H_2$ and

$CO$. A simpler, empirically derived expression called the "cube-root" law is used as the user-subroutine default expression for mixture conductivity [11]:

$$k_{\text{mix}} = \frac{\sum_{i=1}^{n} \sqrt[3]{\mathcal{M}_i} Y_i k_i}{\sum_{i=1}^{n} \sqrt[3]{\mathcal{M}_i} Y_i} \tag{3.73}$$

When the ideal gas property model is disabled, the menu-accessed default expression for mixture thermal conductivity is the mass fraction average of the pure species values:

$$k_{\text{mix}} = \sum_{i=1}^{n} X_i k_i \tag{3.74}$$

#### 3.4.2.5 Binary Mass Diffusivity

**3.4.2.5.1 Binary System** When the ideal gas property model is disabled, the binary mass diffusivity between species $i$ and species $j$, $\mathcal{D}_{ij}$, can be expressed as either a polynomial or piecewise-linear function of temperature.

When the ideal gas property model is enabled, $\mathcal{D}_{ij}$ can be expressed as a polynomial or piecewise-linear function of temperature, or from an expression derived from kinetic theory [10]

$$\mathcal{D}_{ij} = 0.0188 \frac{\left[T^3 \left(\frac{1}{\mathcal{M}_i} + \frac{1}{\mathcal{M}_j}\right)\right]^{\frac{1}{2}}}{P \sigma_{ij}^2 \Omega_{\mathcal{D}}} \tag{3.75}$$

All quantities in equation 3.75 are in SI units except $\sigma_{ij}$ (Å). The dimensionless function $\Omega_{\mathcal{D}}$ depends on $T^*$ where, in the case of binary diffusion:

$$T^* = \frac{T}{(\epsilon/\kappa)_{ij}} \tag{3.76}$$

The relationship between $\Omega_{\mathcal{D}}$ and $T^*$ is taken from [9] and implemented in table format for values of $T^*$ between 0.3 and 100. The values of $\sigma_{ij}$ and $(\epsilon/\kappa)_{ij}$ are estimated from the corresponding Lennard-Jones parameters of the pure species $i$ and $j$ as follows [9]:

$$\sigma_{ij} = \frac{1}{2}(\sigma_i + \sigma_j) \tag{3.77}$$

$$(\epsilon/\kappa)_{ij} = \sqrt{(\epsilon/\kappa)_i (\epsilon/\kappa)_j} \tag{3.78}$$

Note, that calculation of $\mathcal{D}_{ij}$ by the default kinetic theory expression requires input of the Lennard-Jones parameters for both species $i$ and species $j$.

**3.4.2.5.2 Composition Dependent** Binary mass diffusivity applies only to a mixture of two chemical species. The notion of a composition dependent binary diffusivity in a mixture of more than two species is sometimes used, however, in order to approximate the diffusion mass flux, $\mathbf{J}_i$ using Fick's law:

$$\mathbf{J}_i = -\rho \mathcal{D}_{i,\text{mix}} \nabla X_i \tag{3.79}$$

The value of $\mathcal{D}_{i,\text{mix}}$ in equation 3.79 denotes the mass diffusivity of species $i$ diffusing in the local mixture as if the mixture composition were not changing. Allowing $\mathcal{D}_{i,\text{mix}}$ to vary with the local composition can be used to simulate the first order impact of composition variation on $\boldsymbol{J}_i$ without resorting to the more computationally intensive multicomponent transport models.

The expression for $\mathcal{D}_{i,\text{mix}}$ is taken from [11]:

$$\mathcal{D}_{i,\text{mix}} = \frac{1 - Y_i}{\sum_{j=1, i \neq j}^{n} \frac{Y_j}{\mathcal{D}_{ij}}} \tag{3.80}$$

When the mixture is dilute, equation 3.80 reduces to $\mathcal{D}_{i,\text{mix}} = \mathcal{D}_{i,j}$ where species $j$ is the carrier gas. Equation 3.80, therefore, can be considered as an approximation for the effect of non-dilute concentrations.

Equation 3.80 is only used when the dilute option of the species transport equation is chosen.

### 3.4.2.6 Multicomponent Mass Diffusivity

Multicomponent mass diffusivities, $D_{ij}$, are required only for the multicomponent transport model for species flux $\mathbf{J}_i$ (e.g., equation 3.23). For a fluid mixture of $n$ species, $n(n-1)$ independent values of $D_{ij}$ are required. When the dilute transport model is chosen, $D_{ij}$ is not required.

**3.4.2.6.1 Composition Independent** Polynomial or piecewise-linear functions of temperature can be used to calculate the values of $D_{ij}$ when the ideal gas property model is either enabled or disabled. Use of these expressions assumes that all of the $D_{ij}$ are composition independent and that data for all $n(n-1)$ values of $D_{ij}$ will be provided by the user.

**3.4.2.6.2 Composition Dependent** When the ideal gas property model is enabled, a menu-accessed default expression based on kinetic theory is available to evaluate composition dependent $D_{ij}$ [10]:

$$D_{ij} = \frac{\mathcal{M}}{\mathcal{M}_j} \frac{K^{ji} - K^{ii}}{|K|} \tag{3.81}$$

where, $|K|$ is the determinant of the matrix whose $(i,j)$ element is given by:

$$K_{ij} = \frac{Y_i}{\mathcal{D}_{ij}} + \frac{\mathcal{M}_j}{\mathcal{M}_i} \sum_{k=1, k \neq i}^{n} \frac{Y_k}{\mathcal{D}_{ik}} \quad \text{for} \quad i \neq j \tag{3.82}$$

$$K_{ij} = 0 \qquad \text{for} \quad i = j \tag{3.83}$$

and where $K^{ji}$ and $K^{ii}$ are cofactors of matrix $K$.

### 3.2.4.7 Soret Coefficient

When the ideal gas property model is disabled, the thermal diffusion coefficient of species $i, D_i^T$, can be expressed as either a polynomial or piecewise linear function of temperature.

When the ideal gas property model is enabled, $D_i^T$ can be expressed as a polynomial or piecewise-linear function of temperature, or from the following empirical-based composition dependent expression taken from [11]:

$$D_i^T = -2.59(10^{-7})T^{0.659}\left[\frac{\mathcal{M}_i^{0.511}Y_i}{\sum_{i=1}^{n}\mathcal{M}_i^{0.511}Y_i} - X_i\right]\left[\frac{\sum_{i=1}^{n}\mathcal{M}_i^{0.511}Y_i}{\sum_{i=1}^{n}\mathcal{M}_i^{0.489}Y_i}\right] \tag{3.84}$$

Equation 3.84 yields a negative value for species with small molecular weight and a positive value for species with large molecular weight. Use of equation 3.84 in equation 3.23 will therefore impact species diffusion such that heavy molecules diffuse less rapidly, and light molecules diffuse more rapidly, towards heated surfaces.

### 3.4.2.8 Enthalpy of Formation

The enthalpy of formation of species $i, h_{f,i}^{\circ}$, and the corresponding reference temperature $T_{o,i}$, are menu-prompted user inputs required for each species $i$ undergoing chemical reaction. Consistent with the ideal gas assumption, $h_{f,i}^{\circ}$ is a constant value without dependence on pressure.

### 3.4.2.9 Implementation of User-Defined Subroutines For Mixture Properties

As described above, the Phase II effort has provided the capability to "tailor" mixture property evaluation through the use of Fortran subroutines written by the user. Shell subroutines for each property exist (except density) and, further, default expressions for calculating mixture properties have been written and provided with the software. These are the "user-subroutine" default expressions described in sections 3.4.2.2 to 3.4.2.6 and summarized in Appendix C. The user subroutines can be rewritten by the user, however, in order to provide alternate expressions for calculating mixture properties. The remainder of this section describes the steps involved for incorporating a new mixture property expression.

Incorporating a new mixture property relationship is a straightforward task. The first step is to identify the appropriate subroutine as shown in Table 3.5. Then, the default expression currently implemented should be removed. The default expression is bracketed by the comments `BEGIN FUNCTION ..` and `END FUNCTION ...` The recommended approach is to "comment-out" the original FORTRAN source lines and add new FORTRAN source lines which implement the desired mixture property expression. The next step would be to compile the user subroutine and eliminate any coding errors. (The procedure for compiling is system dependent: the user should consult with computer system personnel if the procedure is unknown.)

Once the FORTRAN errors are eliminated, the subroutine is ready to be incorporated into the rest of the code. This is accomplished most easily by replacing the original version of the particular user subroutine, sent with the original installation, with the new version (the user may wish to retain a copy of the original version) and then rebuild the object code version using the procedure described in the installation instructions which were provided with the original program. This procedure will generally overwrite the current executable of FLUENT/CVD. The user may wish to save the original version under a separate name to allow access to the unmodified version.

The new version of the code should be tested to determine if the user-defined expression is behaving as desired. This can be done on simple problems where the calculated property can be computed by hand for comparison.

The procedure described above (coding, compiling, rebuilding, and testing) should be repeated until the user is satisfied that the new mixture property expression is correct.

Table 3.6: **Subroutine Names for User-Defined Mixture Property Calculations**

| Property | Subroutine Name |
|---|---|
| Viscosity | usermu |
| Specific Heat | usercp |
| Thermal Conductivity | usertc |
| Mass Diffusion Coef. | userdm |
| Soret Coefficient | userdt |

## 3.5 PARTICLE TRACKING

At the end of the Phase I project, FLUENT/CVD had the same particle tracking capabilities as the then currently released version of FLUENT, Version 2.99. This included calculation of the trajectories of particles:

- of variable diameter;
- of variable density;
- injected at variable location; and,
- injected with variable vector velocity.

Particles striking solid boundaries could either "disappear" or rebound back into the flow. However, a thermophoretic force on the particle was not available. The workscope of Phase II, therefore, included implementing thermophoresis into the particle tracking capabilities of FLUENT.

### 3.5.1 Equation of Motion for the Particle

Particle motion is treated from a Lagrangian frame of reference; i.e., the frame of reference is attached to the particle. In this case, the equation of motion for the particle is derived from Newton's second law of motion:

$$m\frac{d\mathbf{U}_p}{dt} = \sum \mathbf{F} \tag{3.85}$$

where: $m$ = particle mass
$\mathbf{U}_p$ = particle vector velocity
$t$ = time
$\mathbf{F}$ = vector force acting on the particle

The forces in equation 3.85 are due to interaction between the particle and the fluid including viscous drag, pressure gradients and thermophoresis, as well as body forces such as gravity. Evaluation of the forces in equation 3.85 is based on the following simplifications:

- Stoke's flow, i.e., small value of particle Reynolds number based on particle diameter and relative velocity;
- spherical particles;
- laminar flow;
- steady (non-transient) flow in the fluid;
- negligible particle-particle interaction;
- pressure gradient evaluated from local fluid acceleration, neglecting viscous effects;
- neglect effect of particle acceleration on viscous drag;
- $(\mathbf{U}_p \cdot \nabla)\mathbf{U} \sim (\mathbf{U} \cdot \nabla)\mathbf{U}$ where $\mathbf{U}$ is the local fluid velocity; and,
- the only body force considered is gravity.

With these simplifications the equation of particle motion can be derived from the expression given in [13] as follows:

$$\frac{\pi}{6}D_p^3\rho_p\frac{d\mathbf{U}_p}{dt} = \mathbf{F}_D + \mathbf{F}_T + \frac{1}{2}\,\frac{\pi}{6}D_p^3\rho\left[(\mathbf{U}\cdot\nabla)\mathbf{U} - \frac{d\mathbf{U}_p}{dt}\right]$$

$$+\frac{\pi}{6}D_p^3\rho\left[(\mathbf{U}\cdot\nabla)\mathbf{U}\right] + \frac{\pi}{6}D_p^3(\rho_p - \rho)\mathbf{g} \tag{3.86}$$

where: $\mathbf{F}_D$ = drag force, vector
$\mathbf{F}_T$ = thermophoretic force, vector
$D_p$ = particle diameter
$\rho_p$ = particle density
$\rho$ = local fluid density
$\mathbf{g}$ = gravity vector

The third term on the right hand side (RHS) of equation 3.86 represents the force required to accelerate the virtual mass of the fluid. The fourth term on the RHS derives from the pressure gradient in the fluid. The last term on the RHS represents the gravitational body force. The drag force, $\mathbf{F}_D$, is evaluated from Stoke's Law for drag on a spherical particle:

$$\mathbf{F}_D = 3\pi\mu D_p(\mathbf{U} - \mathbf{U}_p) \tag{3.87}$$

Equation 3.86 without $\mathbf{F}_T$ and with $\mathbf{F}_D$ given by 3.87 is the particle tracking equation used by FLUENT/CVD at the end of Phase I. Phase II work consisted of implementing $\boldsymbol{F}_T$ into the particle tracking equation. The expression used for $\mathbf{F}_T$ is described in the next section.

### 3.5.2 Thermophoretic Force

The expression for the thermophoretic force on a particle is taken from [14]:

$$\mathbf{F}_T = \frac{6\pi\mu^2 D_p C_S(K + C_t Kn)}{\rho(1 + 3C_m Kn)(1 + 2K + 2C_t Kn)} \frac{\nabla T}{T} \tag{3.88}$$

where: $Kn$ = $2\,\lambda/\mathrm{D}_p$
$\lambda$ = mean-free-path of the fluid
$K$ = $k/k_p$
$k$ = fluid thermal conductivity
$k_p$ = particle thermal conductivity
$C_S$ = 1.17
$C_t$ = 2.18
$C_m$ = 1.14
$T$ = local fluid temperature
$\mu$ = fluid viscosity

This expression assumes that the particle is a sphere and that the fluid is an ideal gas. The mean-free-path of the gas used in equation 3.88 is based on the local viscosity:

$$\lambda = \sqrt{\frac{\pi}{2}} \frac{\mu}{P}\sqrt{RT} \tag{3.89}$$

where: $P$ = local gas pressure
$R$ = universal gas constant

Also, the thermal conductivity of the gas used in equation 3.88 is based on translational energy only [14]:

$$k = \frac{15}{4}\mu R \tag{3.90}$$

The impact of particle size on the thermophoretic force predicted by equation 3.88 depends on the value of the Knudsen number, $Kn$. When the particle is large compared to the mean-free-path, $Kn << 1$ and:

$$\mathbf{F}_T(Kn << 1) = -7.02\pi\frac{\mu^2}{\rho}D_p\frac{K}{1+K}\frac{\nabla T}{T} \tag{3.91}$$

Thus, for large particles, the thermophoretic force is affected by the thermal conductivity of the particle. In many CVD applications, the particles are small, being nucleated in the process gas, and the gas pressure is low providing a large mean-free-path. In these cases, the Knudsen number is large and:

$$\mathbf{F}_T(Kn >> 1) = 0.51\pi\frac{\mu^2}{\rho}\frac{D_p^2}{\lambda}\frac{\nabla T}{T} \tag{3.92}$$

For small particles, therefore, the effect of particle conductivity is not important.

# 4 DEMONSTRATION OF SOFTWARE ENHANCEMENTS

Many numerical simulations were performed during the Phase II effort to verify and demonstrate the various enhancements implemented in this project. These include:

- repeat of Phase I results;
- verification of kinetic theory and mixture property calculations;
- separate effects tests of physical models to verify implementation; and,
- demonstration of modeling capabilities on realistic CVD simulations.

Results obtained with the Phase II enhanced versions of FLUENT reproduce the results of Phase I. Verification testing of physical models, including property calculations, were successful. Since the physical models are incorporated into the marketed FLUENT software, the proper implementation of the physical models will be maintained under normal FLUENT development by Fluent Inc. This section presents examples of FLUENT results in order to illustrate the new simulation capabilities for realistic CVD configurations.

The results of seven FLUENT simulations are described in the following sections. They consider a horizontal reactor configuration for the deposition of silicon from silane. The models build in complexity in a way that is usually beneficial for simulating complex phenomena. Table 4.1 lists the demonstration cases with a brief description of the characteristics of the model.

All results shown in the following sections were obtained using the released version of FLUENT V3.03. This version has been delivered to NASA/Langley at the time of submission of this report.

Table 4.1: **Summary of Sample Cases**

| Case | Characteristics |
|---|---|
| A | Baseline. Deposition on heated substrate. Single surface reaction: $SiH_4 \rightarrow Si(s) + 2H_2$ |
| B | Case A with the addition of heat conduction in the lower quartz walls |
| C | Case B with a three reaction mechanism, gas phase reaction, and mass diffusivity calculated by kinetic theory |
| D | Case C with deposition on quartz |
| E | Case D with thermal diffusion |
| F | Case E in three dimensions, with and without natural convection. |
| G | Transient version of Case A |

## 4.1 CASE A – BASELINE HORIZONTAL REACTOR

Case A considers the horizontal reactor shown in Figure 4.1. The geometry is two-dimensional in the cartesian coordinate system. The inlet gas mixture enters the reactor at the inlet and for 5 cm flows between quartz walls which are separated by 2 cm. After this inlet region, the gas mixture then flows over the heated substrate; the length of the heated region is 10 cm. A sudden expansion from 2 cm to 3.5 cm occurs in the flow channel beyond the heated substrate. Recirculation of the flow can be expected to occur at this backward-facing step.

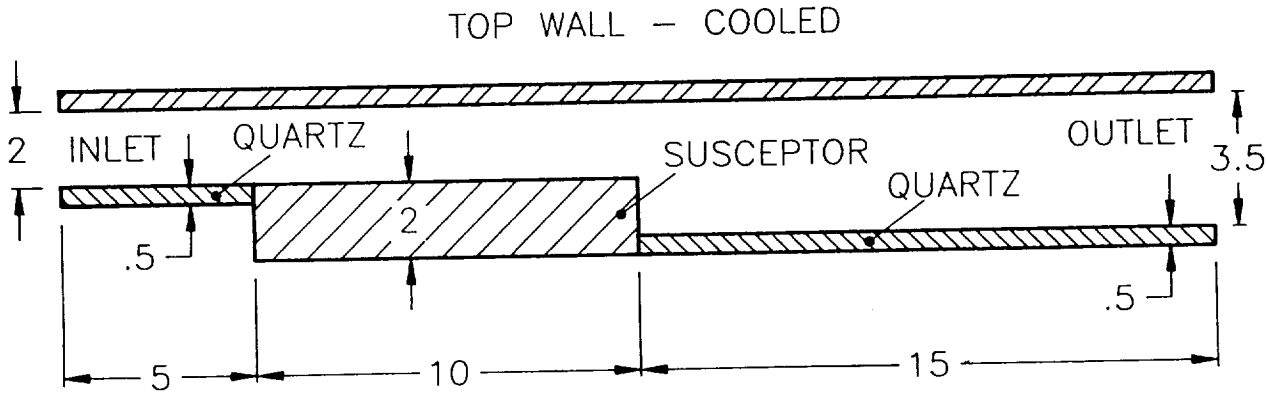


Figure 4.1: Baseline Horizontal Reactor Configuration. (All dimensions in centimeters.)

Deposition of silicon from silane, $SiH_4$, is examined in this horizontal reactor. For the baseline case, the following single-step surface deposition reaction is simulated:

$$SiH_4(g) \rightarrow Si(s) + 2H_2(g) \tag{4.1}$$

In particular, the deposition reaction is taken to be diffusion controlled. That is, any silane which diffuses to solid surfaces will react and deposit silicon. A diffusion controlled reaction is simulated by chosing reaction rate constants which produce a very large reaction rate; it is verified by inspection of the solution to check that the concentration of $SiH_4$ at the deposition surfaces is close to zero.

Boundary conditions for the baseline model are as follows:

**Inlet.** The inlet gas is a mixture of $SiH_4$ and hydrogen ($H_2$). The mass fraction of $SiH_4$ in the inlet mixture is 0.0157; the remainder is $H_2$. The inlet temperature is 300 K and the inlet velocity is parabolic with zero velocity at the wall and average velocity of 17.5 cm/s.

**Walls.** All walls have the no-slip or zero velocity boundary condition. The quartz walls are cooled to 300 K. The susceptor is heated to a uniform temperature of 1300 K. Deposition is allowed only on the susceptor. Reaction (4.1) is disabled for the quartz walls via menu-prompted input. This demonstrates the capability to simulate surface reactions which have varying reaction rates on different surfaces (e.g., catalyzed by a particular surface in the reactor).

The gas mixture is treated as dilute. The gas properties are taken to be those of pure $H_2$ and are temperature dependent:

$$\underline{\text{Density}}$$
Ideal gas with molecular weight 2.016 at 1 atmosphere pressure

$$\underline{\text{Viscosity (kg/m-s)}}$$
$$3.17(10^{-6}) + 2.04(10^{-8})T - 3.52(10^{-12})T^2$$

$$\underline{\text{Specific Heat (J/kg-K)}}$$
Piecewise Linear: 14191 at 255.6 K
15739 at 1367 K

$$\underline{\text{Thermal Conductivity (w/m-k)}}$$
$$3.80(10^{-2}) + 5.41(10^{-4})T - 2.15(10^{-7})T^2 + 8.57(10^{-11})T^3$$

$$\underline{\text{Mass Diffusivity of } SiH_4 \text{ (m}^2\text{/s)}}$$
$$7.234(10^{-8})T + 4.659(10^{-10})T^2 - 8.016(10^{-14})T^3$$

Some or all of these properties could have been calculated by kinetic theory. It is preferred, however, to use tabulated actual data when it is available, as was done here for viscosity, specific heat and thermal conductivity. The polynomial expression for the mass diffusivity of

$SiH_4$ was used to maintain correspondence with Phase I models which did not have kinetic theory calculation capabilities.

A grid of 62 grid points in the axial direction and 30 grid points in the cross-stream direction was used. Figure 4.2 shows the resulting control volumes with the vertical scale magnified by a factor of four. The grid size is refined near the leading edge of the susceptor where gradients in temperature and species concentration are expected to be largest.

Velocity vectors for the baseline case are shown in Figure 4.3. Note that the magnitude of velocity increases as the gas flows over the heated susceptor. As the gas temperature increases, its density becomes less and the velocity increases to conserve the mass flow rate. Also, a small recirculation zone exists beyond the susceptor at the sudden expansion in the channel width.

Figures 4.4 and 4.5 show contours of constant temperature and silane concentration, respectively. In both figures, the gradients are seen to be large at the leading edge of the susceptor where the wall temperature changes abruptly and where the deposition reaction begins. As expected from the diffusion controlled surface deposition mechanism used in this model, the concentration of $SiH_4$ near the susceptor is very close to zero. The concentration gradient of $SiH_4$ above the susceptor produces diffusion of $SiH_4$ above the susceptor to the surface which feeds the surface reaction (4.1).

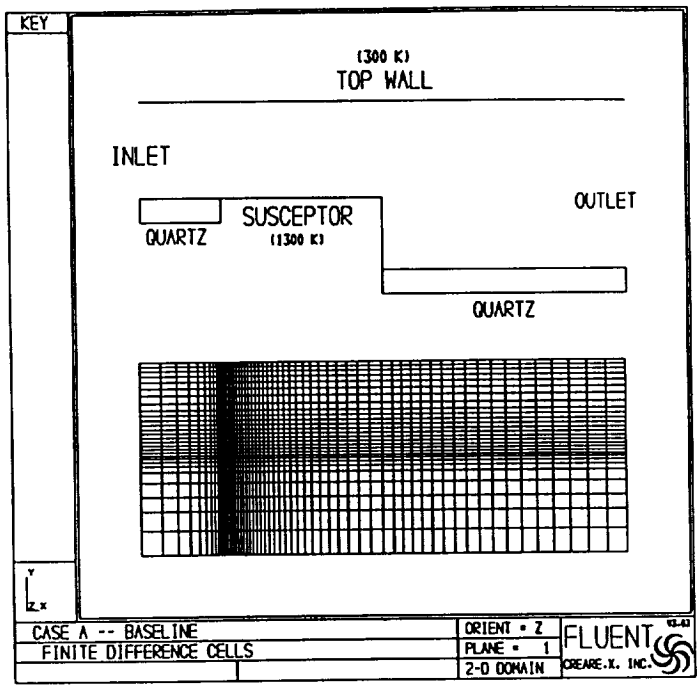


Figure 4.2: Computation Cells for the Baseline Case

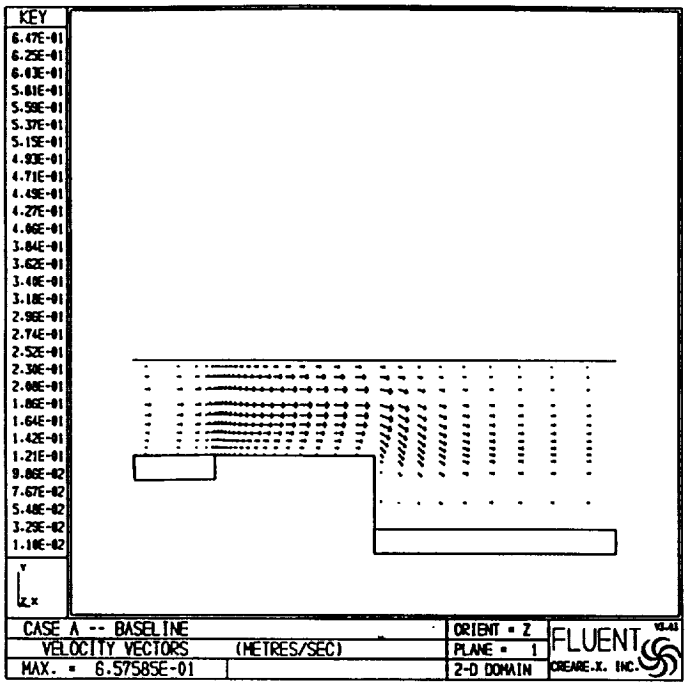


Figure 4.3: Velocity Vectors for the Baseline Case (Case A)

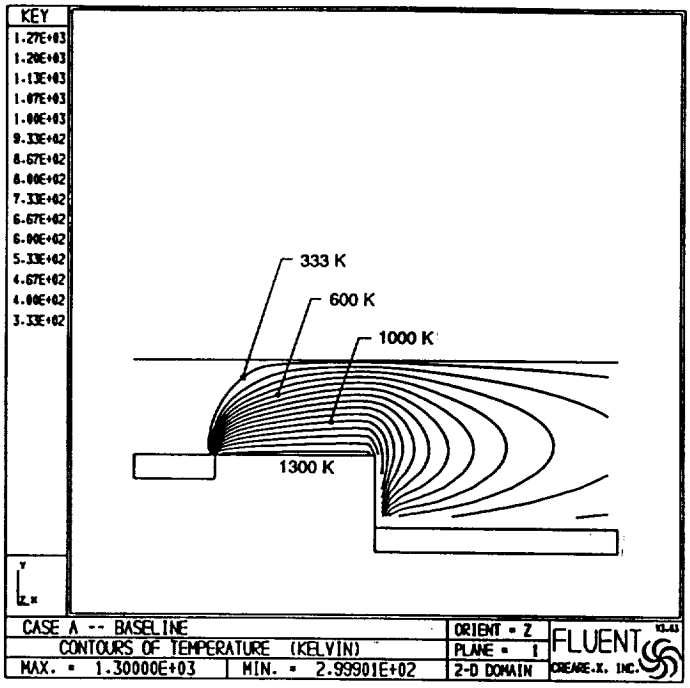


Figure 4.4: Temperature Contours for the Baseline Case (Case A)

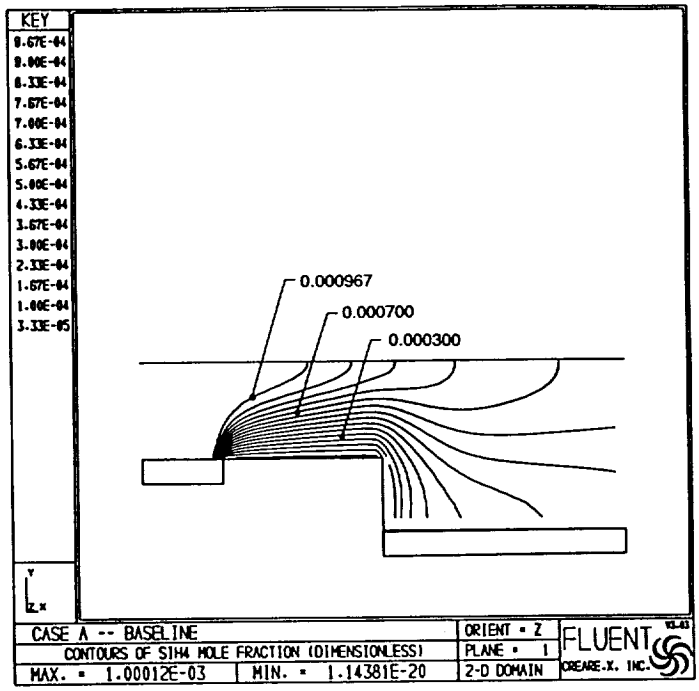


Figure 4.5: $SiH_4$ Concentration Contours for the Baseline Case (Case A)

The solid line of Figure 4.6 shows the deposition rate vs. axial position. The deposition rate is quite large near the leading edge of the susceptor and decreases with distance from the leading edge, as is typical for boundary layer flows. The deposition rate, however, falls more rapidly than would be expected from a pure boundary layer flow (i.e., more rapidly than $\sqrt{1/x}$) because, as seen in Figure 4.5, the bulk gas mixture is becoming depleted in $SiH_4$ before reaching the end of the susceptor.

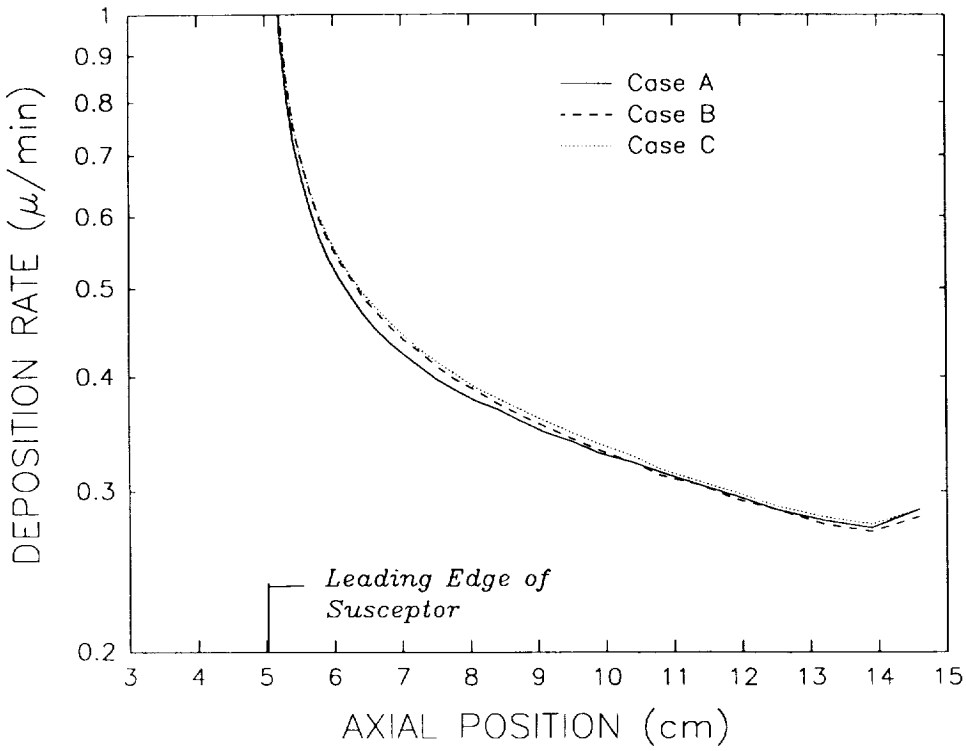


Figure 4.6: Deposition Rate for Cases A, B and C

## 4.2 CASE B – ADDITION OF HEAT CONDUCTION IN LOWER QUARTZ WALLS

This case is the same as Case A (Baseline) except that the lower quartz walls are not modeled at constant temperature but are modeled as thermally conducting. Heat will conduct into these quartz walls from the edges that are in contact with the 1300 K susceptor.

The exterior edges of the quartz walls are taken to be adiabatic (zero heat flux). Thus, the heat conducted from the susceptor eventually is transferred to the gas mixture through the interior faces of the lower quartz walls.

The thermal conductivity of the quartz is given by the following polynomial expression:

$$R_{quartz}(w/m-k) = 1.692 - 0.00193T + 3.196(10^{-6})T^2$$

The upper quartz wall remains cooled at 300 K.

The velocity vectors and contours of $SiH_4$ concentration for Case B are very similar to those of Case A. The temperature contours of Figure 4.7 show, however, that heat conduction into the quartz wall heat the gas mixture upstream of the susceptor. The overall impact on deposition rate is small as seen in Figure 4.6 (dashed line). The deposition rate for Case B is

slightly larger than that of Case A for the first few centimeters of the susceptor. This is due to the higher temperature of the gas mixture which causes the mass diffusivity of $SiH_4$ to increase. Since the deposition is diffusion controlled, the deposition rate therefore increases.

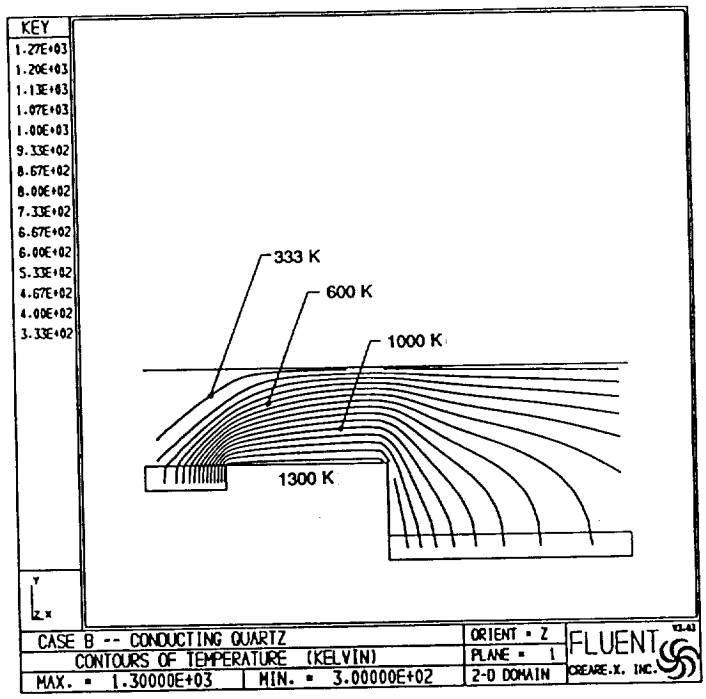


Figure 4.7: Temperature Contours for Case B

## 4.3 CASE C – ADDITION OF MULTIPLE REACTIONS

Case C is the same as Case B except that the single-step surface reaction (4.1) is replaced by the following three reaction mechanism:

$$SiH_4(g) \rightarrow SiH_2(g) + H_2(g) \tag{4.2a}$$

$$SiH_4(g) \rightarrow Si(s) + 2H_2(g) \tag{4.2b}$$

$$SiH_2(g) \rightarrow Si(s) + H_2(g) \tag{4.2c}$$

Reaction (4.2a) is a gas-phase reaction which breaks $SiH_4$ into $SiH_2$. The reaction rates for this reaction are not used as a wall boundary condition; rather, this reaction contributes to the source term of the species conservation equation (e.g., equation 3.10). Also note that $Si(s)$ is deposited by two separate surface reactions; both $SiH_4$ and $SiH_2$ are modeled as sources for $Si(s)$ deposition in this reaction mechanism.

The reaction rates used for reaction mechanism (4.2) are taken from Coltrin, et al. [15]. Surface deposition from $SiH_2$, reaction (4.2c) is taken to be diffusion controlled; i.e., the

reaction rate is very large and any $SiH_2$ which diffuses to the surface reacts and deposits $Si(s)$. The reaction rate for the gas phase reaction (4.2a) is provided in Arrhenius form in [15]; with appropriate adjustment for consistency of units, the Arrhenius reaction rate constants (c.f., equation 3.45a) are):

$$\begin{aligned} A &= 2.54(10^{38})^1 \\ \beta &= -7.95 \\ E &= 2.59(10^8) \\ \nu_{SiH_4} &= 1 \end{aligned} \tag{4.3}$$

The reaction rate for surface deposition from $SiH_4$, reaction (4.2b) is derived from kinetic theory considerations as the rate of arrival of $SiH_4$ molecules to the surface times a reaction rate probability, $\gamma$:

$$R_{SiH_4}(kg/m^2 - s) = \left(\frac{k_B T}{2\pi M_{SiH_4}}\right)^{\frac{1}{2}} \gamma M_{SiH_4} C_{SiH_4} \tag{4.4}$$

where $k_B$ is the Boltzman's constant, 7826 J/kgmol-K, and $C_{SiH_4}$ is the mole concentration of $SiH_4$ in the units of kgmol/m$^3$. The reaction probability is given in [15] as:

$$\gamma = 0.0537e^{-9400/T} \tag{4.5}$$

With appropriate rearrangement, equations 4.4 and 4.5 can be reformulated into FLUENT's standard Arrhenius expression with the following constants:

$$\begin{aligned} A &= 0.334 \\ \beta &= 1/2 \\ E &= 78.15(10^6) \\ \nu_{SiH_4} &= 1 \end{aligned} \tag{4.6}$$

With the addition of $SiH_2$ to the gas mixture, the model must include information to calculate the mass diffusivity of $SiH_2$. Since data for mass diffusivity of $SiH_2$ in the hydrogen carrier gas is not available, Case C uses the kinetic theory expressions, equation 3.75. The kinetic theory parameters for $H_2$ are [9]:

[1]FLUENT does not accept a number as large as $A$ in equation 4.3. To implement this reaction rate, a user-defined reaction rate subroutine was used to "absorb" some of $A$ into the temperature dependency as follows:

$$Rate\left[\frac{kgmole\ SiH_4}{m^3 - s}\right] = 3.588(10^{14})\left(\frac{T}{1000}\right)^{-7.95} C_{SiH_4}\ exp\left[\frac{-2.59(10^8)}{RT}\right]$$

| | Hydrogen |
|---|---|
| $\sigma$ | $= 2.915$ Å |
| $\epsilon/\kappa$ | $= 38$ K |

and for $SiH_2$ [15]:

| | $SiH_2$ |
|---|---|
| $\sigma$ | $= 3.803$ Å |
| $\epsilon/\kappa$ | $= 133.1$ K |

The fluid velocity pattern and temperature field for Case C is very similar to those of Case B. The principal effect of changing the reaction mechanism is the distribution of species concentration. Figures 4.8 and 4.9 show the distributions of $SiH_2$ and $SiH_4$, respectively, for Case C. As Figure 4.8 illustrates, $SiH_2$ begins to be produced in the gas phase only after the gas temperature increases. In particular, since conduction in the quartz tends to preheat the gas (c.f., Figure 4.7), $SiH_2$ production commences upstream of the susceptor. And since, according to the model for Case C, deposition does not occur on the quartz, the concentration of $SiH_2$ builds up along the quartz surface up to the leading edge of the susceptor. Thus, there is a maximum of $SiH_2$ concentration at this location.

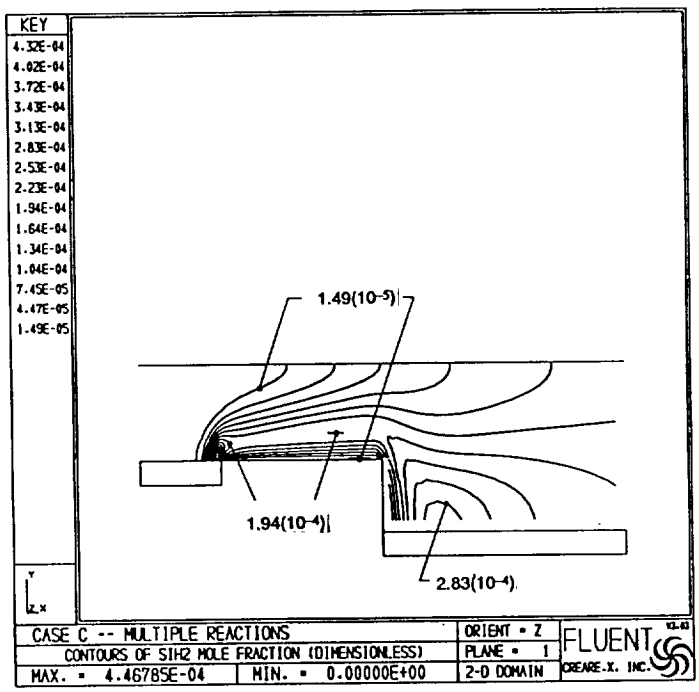


Figure 4.8: $SiH_2$ Concentration Contours for Case C

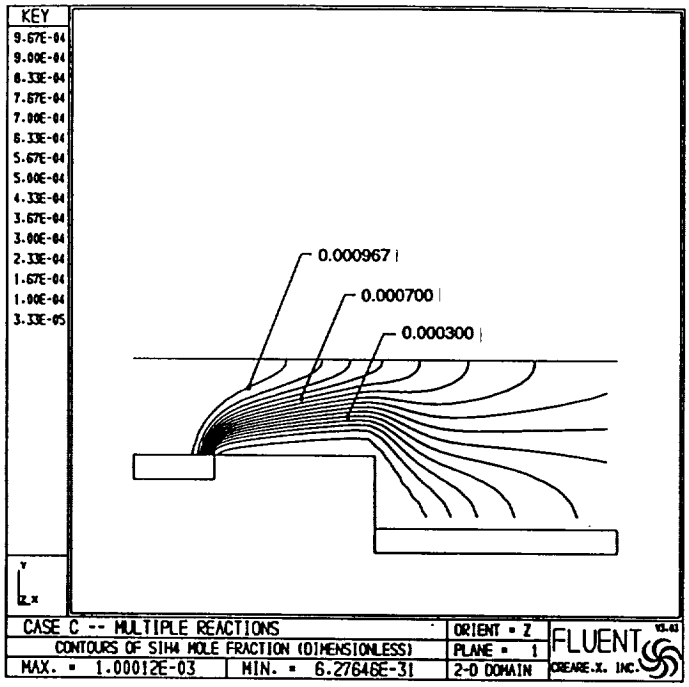


Figure 4.9: $SiH_4$ Concentration Contours for Case C

Above the susceptor, $SiH_2$ diffuses to the surface and deposits solid silicon according to reaction (4.2c). The deposition reaction is assumed to be diffusion limited; the concentration of $SiH_2$ therefore goes to zero at the surface. The source of the $SiH_2$ continues to be the gas phase reaction (4.2a). However, the rate of the gas phase reaction drops off with distance from the susceptor as the gas temperature decreases. This balance between production and depletion of $SiH_2$ yields another local maximum in $SiH_2$ composition about 1/2 cm above the susceptor surface.

The reaction kinetics of reaction (4.2b) is fast at the susceptor temperature of 1300 K. This is apparent in Figure 4.9 because the concentration of $SiH_4$ approaches zero at the surface of the susceptor. Thus, both silicon bearing species ($SiH_2$ and $SiH_4$) deposit silicon on the surface from diffusion controlled reactions and the deposition rate should not be significantly affected by the exchange of $SiH_4$ for $SiH_2$ in the gas phase. This is a principal conclusion of [16]. The deposition rate profiles for Cases B and C in Figure 4.6 demonstrate this conclusion.

## 4.4 CASE D – ADDITION OF DEPOSITION ON QUARTZ

Case D is identical to Case C except that deposition is allowed to occur on the lower quartz walls upstream and downstream of the susceptor. The principal impact of this modeling change is in the distribution of $SiH_2$. Compared to the results of Case C (Figure 4.8), Figure 4.10 shows that the concentration of $SiH_2$ no longer has the local maximum at the

leading edge of the susceptor. Removal of $SiH_2$ on the surface of the quartz prevents the buildup of gas phase $SiH_2$ upstream of the susceptor as occurred for Case C.

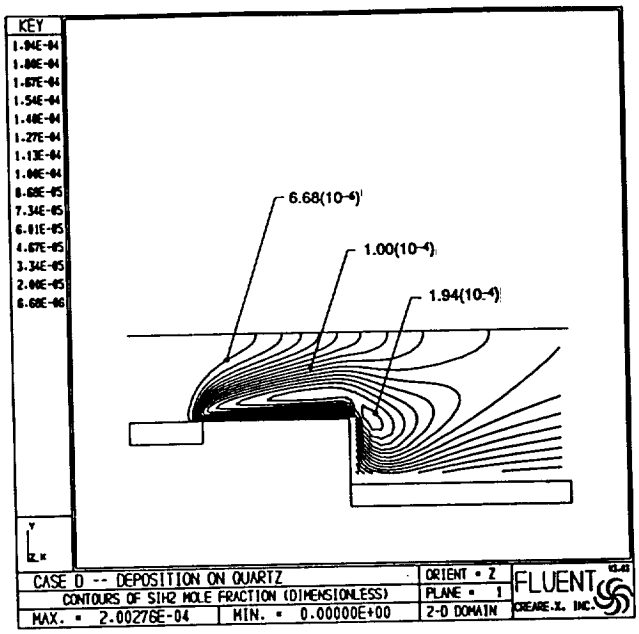


Figure 4.10: $SiH_2$ Concentration Contours for Case D

Figure 4.11 compares the predicted silicon deposition rates for Cases C and D. The deposition rate is still a maximum at the leading edge. However, the singularity in deposition rate at the leading edge of the susceptor is eliminated by allowing upstream deposition on the quartz.

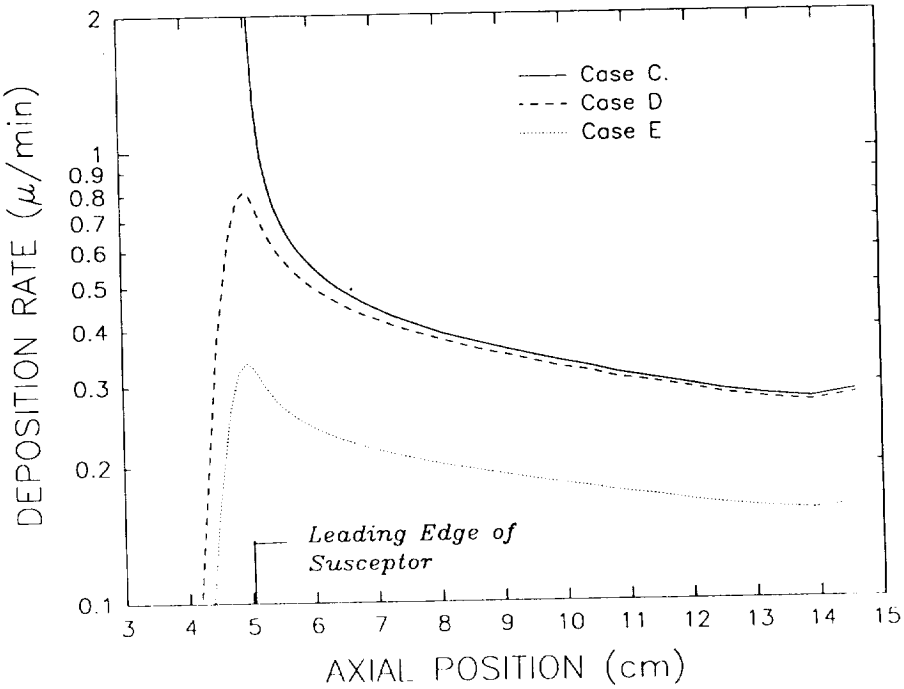


Figure 4.11: Deposition Rate for Cases C, D and E

## 4.5 CASE E – ADDITION OF THERMAL DIFFUSION

Case E is the same as Case D except that thermal (Soret) diffusion of chemical species is included in the model. The thermal diffusion coefficients are calculated by FLUENT according to the default expression, equation 3.84.

The principal impact of thermal diffusion predicted by FLUENT is in the distribution of chemical species. Since the $SiH_2$ and $SiH_4$ molecules are both significantly larger than the hydrogen carrier gas, thermal diffusion will tend to push them away from the heated susceptor. The concentration profiles for $SiH_2$ and $SiH_4$ shown in Figures 4.12 and 4.13, respectively, illustrate this behavior. (Compare to Figures 4.10 and 4.9 without thermal diffusion.) Local maxima are evident at the cold top wall of the reactor. Additionally, the gradients of concentration at the surface of the susceptor are decreased; for this diffusion controlled deposition, the deposition rate also decreases. Figure 4.11 shows that thermal diffusion reduces the deposition rate by a factor of about 1.7.

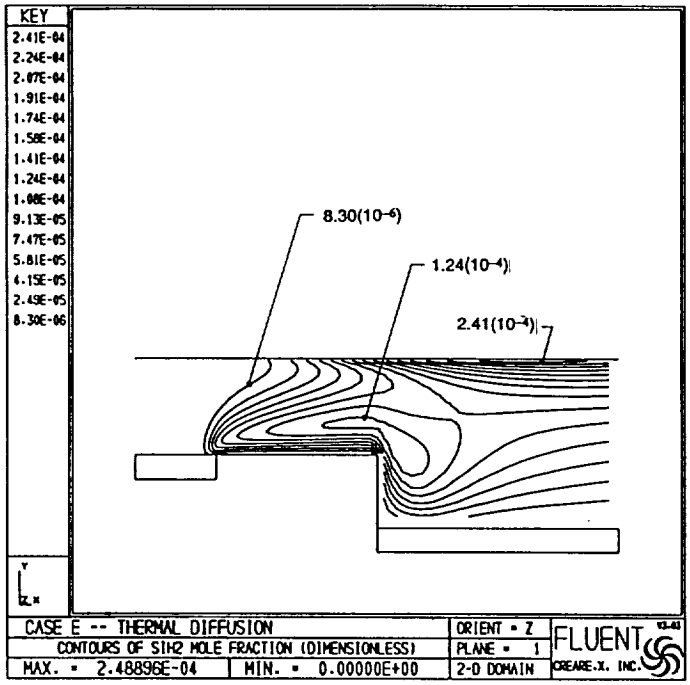


Figure 4.12: $SiH_2$ Concentration Contours for Case E

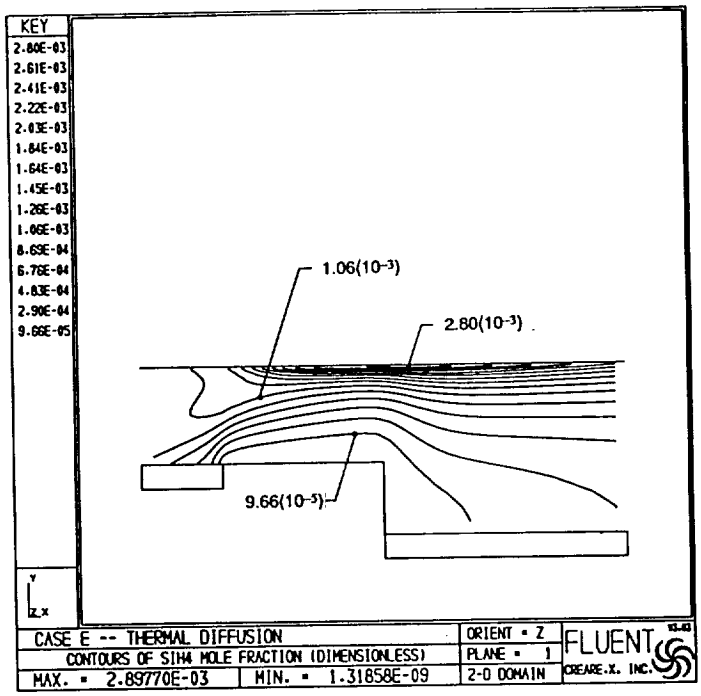


Figure 4.13: $SiH_4$ Concentration Contours for Case E

## 4.6 CASE F – THREE-DIMENSIONAL MODEL

Case F considers a three-dimensional version of Case E. The third dimension is the width of the reactor as viewed from above the reactor in Figure 4.1. The width of the gas domain within the reactor is 5 cm. Walls of quartz, 0.5 cm thick, enclose the sides of the reactor. The side walls are modeled as thermally conducting walls. Thus, the side walls conduct heat from the heated susceptor and lower quartz walls upward toward the cooled top wall. In addition, deposition of silicon is allowed on the side walls.

Case F is symmetric about the mid-plane of the reactor in the width direction. Only one-half of the reactor, therefore, needs to be modeled. Thirteen gird points were used in the width direction, bringing the total number of grid points to 24, 180 (62 axial $\times$ 30 length $\times$ 13 width).

Figure 4.14 shows the deposition rate of silicon for Case F at various positions in the width dimension from the centerline to the sidewall. It is readily apparent that the deposition rate is much larger at the centerline compared to near the sidewall. There are two factors which contribute to this behavior. First, as shown in Figure 4.15, the temperature of the sidewall is larger than the gas adjacent to the sidewall. Thus, thermal diffusion in this region will tend to push the reactant species away from the sidewall and toward the centerline. Second, deposition on the sidewall depletes the gas mixture of reactant for deposition on the susceptor near the sidewall. Both of these effects will reduce the concentration gradients toward the susceptor near the sidewall, as shown in Figure 4.16 for $SiH_4$. And since the surface reaction rates are diffusion controlled, the reduced gradients lead to reduced deposition rate.

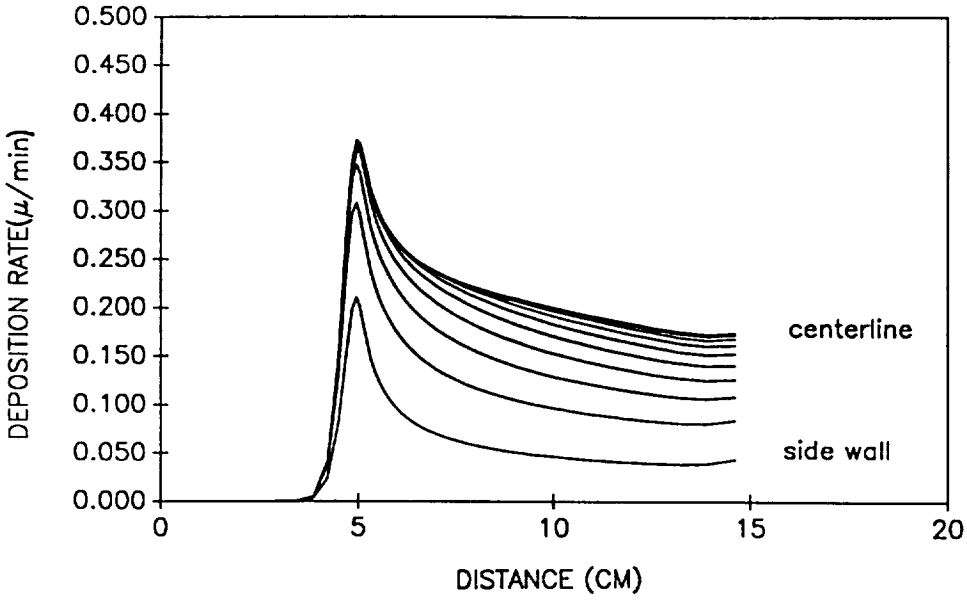


Figure 4.14: Deposition Rate for Case F Without Natural Convection

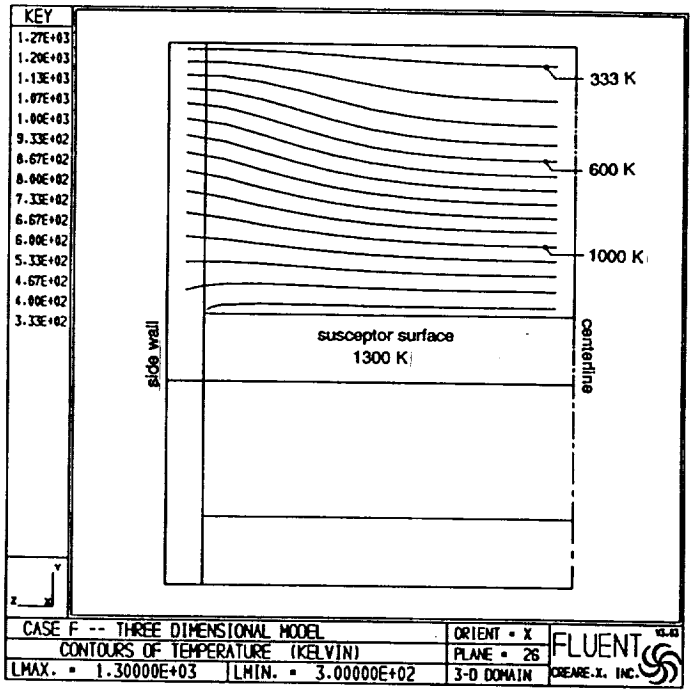


Figure 4.15: Temperature Contours for Case F. End View Looking in the Direction of Flow 2 cm from the Leading Edge of Susceptor

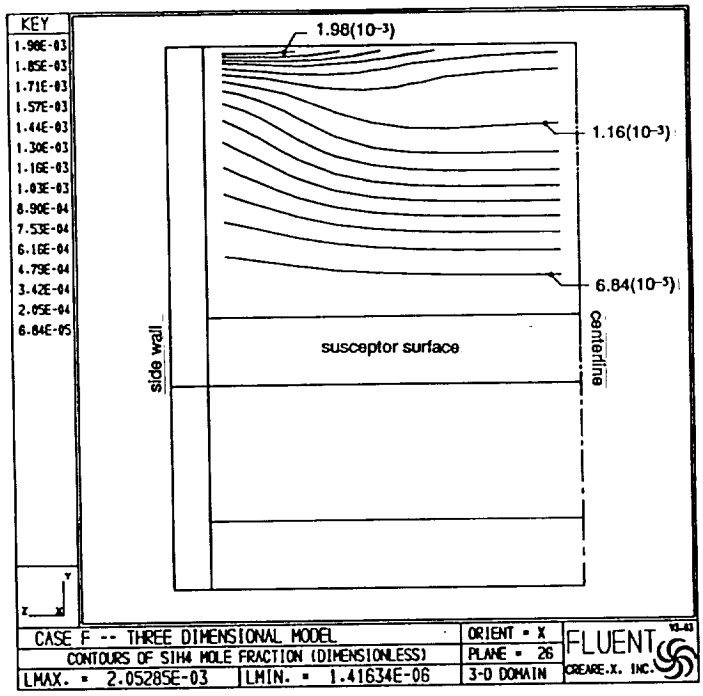


Figure 4.16: Contours of $SiH_4$ Concentration for Case F. End View Looking in the Direction of Flow 2 cm from the Leading Edge of Susceptor

The results described above for Case F do not include natural convection as part of the model. This was accomplished by setting the value of the gravity vector to zero in the model definition. Case F was modified to include natural convection by using 9.8 m/s$^2$ for the magnitude of the gravity vector with a direction pointing toward the bottom of the reactor.

Figure 4.17 shows velocity vectors in a plane 2 cm from the front of the susceptor. Natural convection is seen to generate a recirculating flow pattern in this plane in a direction such that the flow rises along the sidewall. As described above with reference to Figure 4.15, heat conduction in the sidewall increases the gas temperature near the sidewall compared to the centerline. The hotter, less dense fluid along the sidewall thus rises. If the sidewall were cooled, however, one would expect that natural convection would cause the flow pattern to circulate in the opposite sense.

The recirculating flow tends to bring fresh reactant gases from the top of the reactor to the susceptor. Comparing the deposition rates shown in Figure 4.14 (no natural convection) and Figure 4.18 (with natural convection), the recirculation is seen to increase the deposition rate. The effect is small, however, since the maximum velocity of the recirculating flow is not large, about 2 to 3 cm/s or about 10% of the maximum axial velocity.

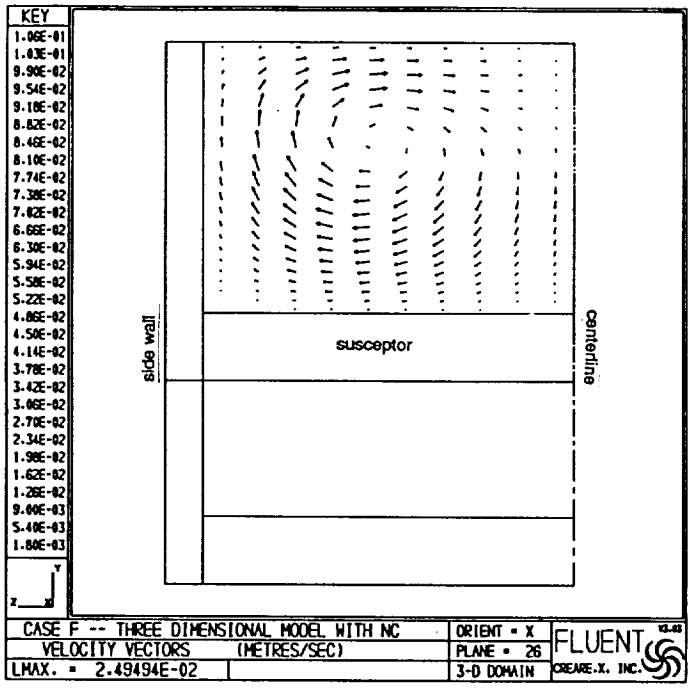


Figure 4.17: Velocity Vectors for Case F Including Natural Convection. End View Looking in the Direction of Flow 2 cm From the Leading Edge of the Susceptor

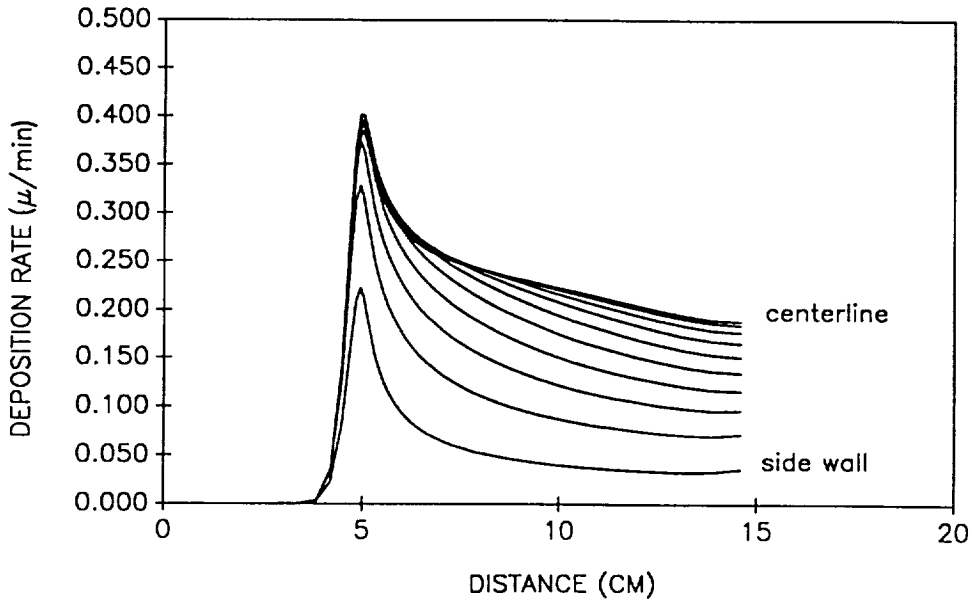


Figure 4.18: Deposition Rate for Case F Including Natural Convection

## 4.7 CASE G – TRANSIENT VERSION OF BASELINE CASE

This case illustrates transient calculations using FLUENT. A transient model was constructed by utilizing the steady-state solution of the baseline case (Case A) as an initial condition with a sudden change of inlet gas composition to pure hydrogen at time equal to zero. This transient model is Case G.

Transient calculations are performed by FLUENT as a sequence of time steps. Iterations are performed within each time step until convergence is achieved. The transient values of the solution parameters are obtained from these converged solutions. Case G used a time step of 0.01 seconds at the start of the transient up to 0.06 seconds, and a time step of 0.02 seconds thereafter.

Figures 4.19a and 4.19b shows $SiH_4$ concentration profiles at various times within the transient, up to 1.0 seconds. The initial condition, t=0, is shown in Figure 4.12. It is evident that the $SiH_4$ concentration begins to be appreciably affected above the susceptor only after about 0.2 seconds. This is approximately the time required for the inlet gas of new composition to travel from the inlet to the leading edge of the susceptor. After about 0.2 seconds, the $SiH_4$ begins to be swept from above the susceptor toward the exit. The $SiH_4$ lingers at the walls, however, because of the reduced velocity near the walls. This is especially apparent at the cold, top wall where $SiH_2$ and $SiH_4$ are not removed by surface reaction.

Figure 4.20 shows the progression with time of the deposition rate. Again, it is seen that about 0.2 seconds are required before the change in inlet composition is observed at the susceptor. After 0.2 seconds, the deposition rate drops rapidly as the $SiH_4$ near the susceptor is utilized without replenishment from the faster moving fluid at the center of the reactor.

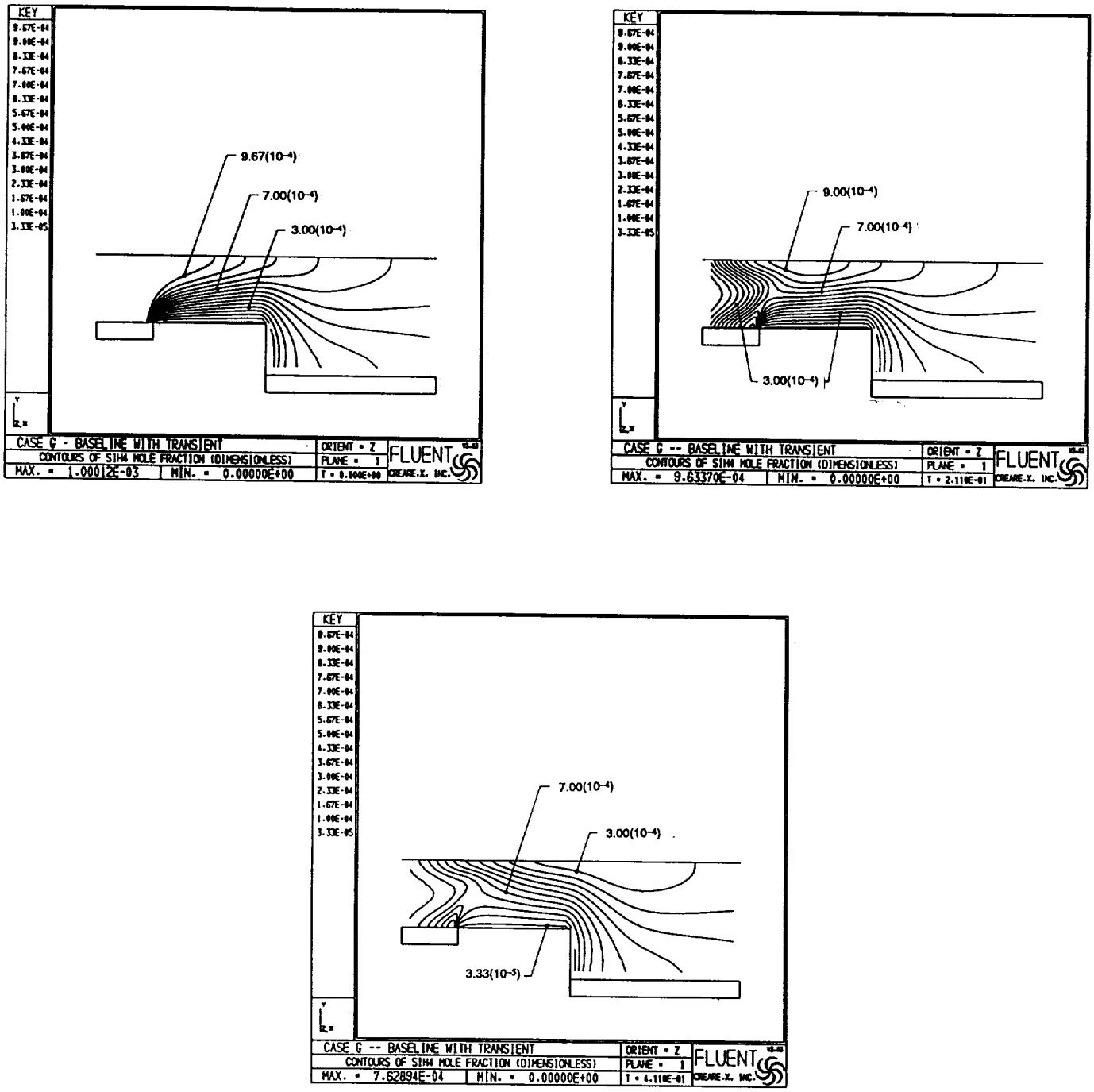


Figure 4.19a: $SiH_4$ Concentration Contours for Case F: 0, 0.2 and 0.4 Seconds

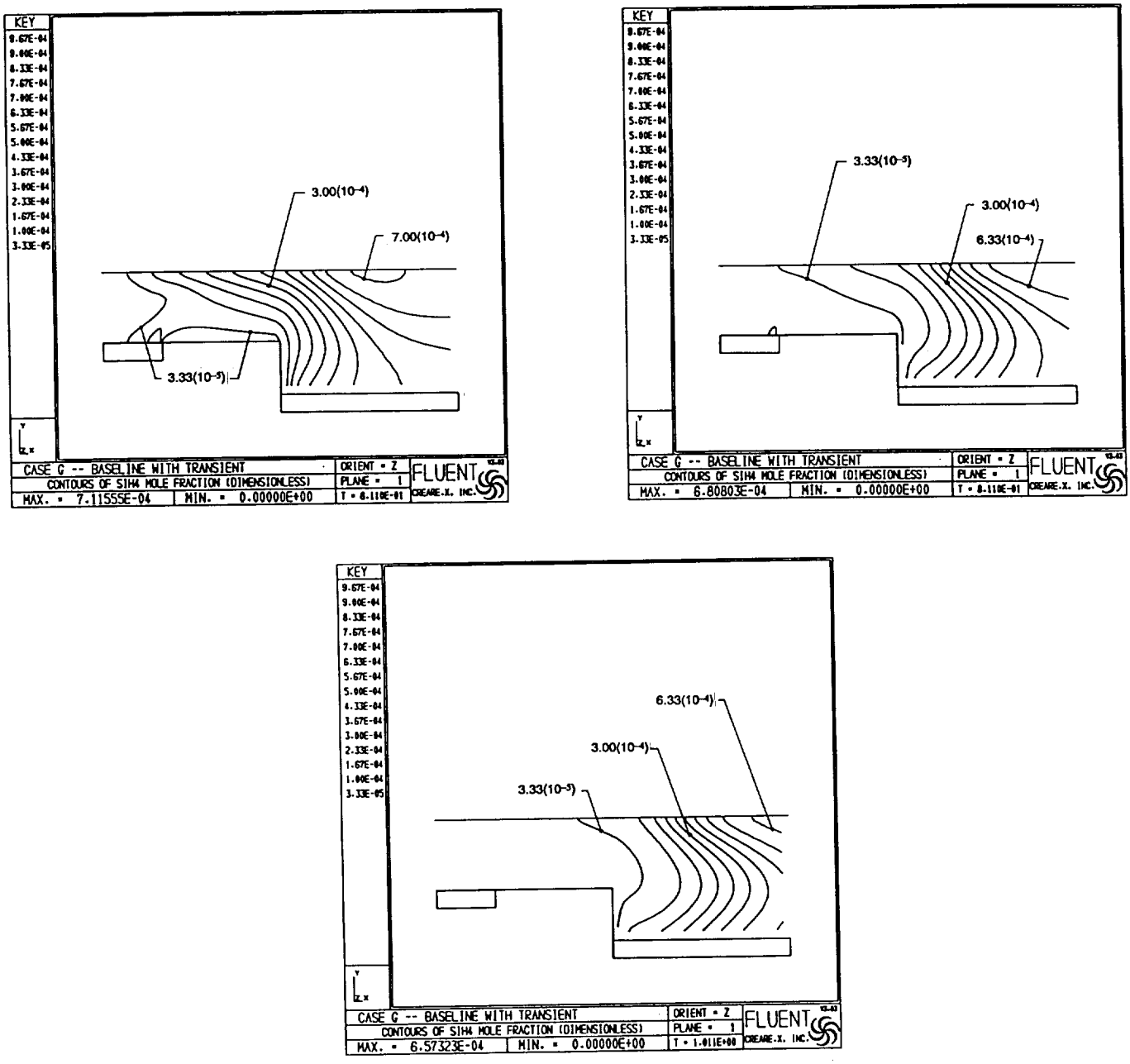


Figure 4.19b: $SiH_4$ Concentration Contours for Case F: 0.6, 0.8 and 1.0 Seconds

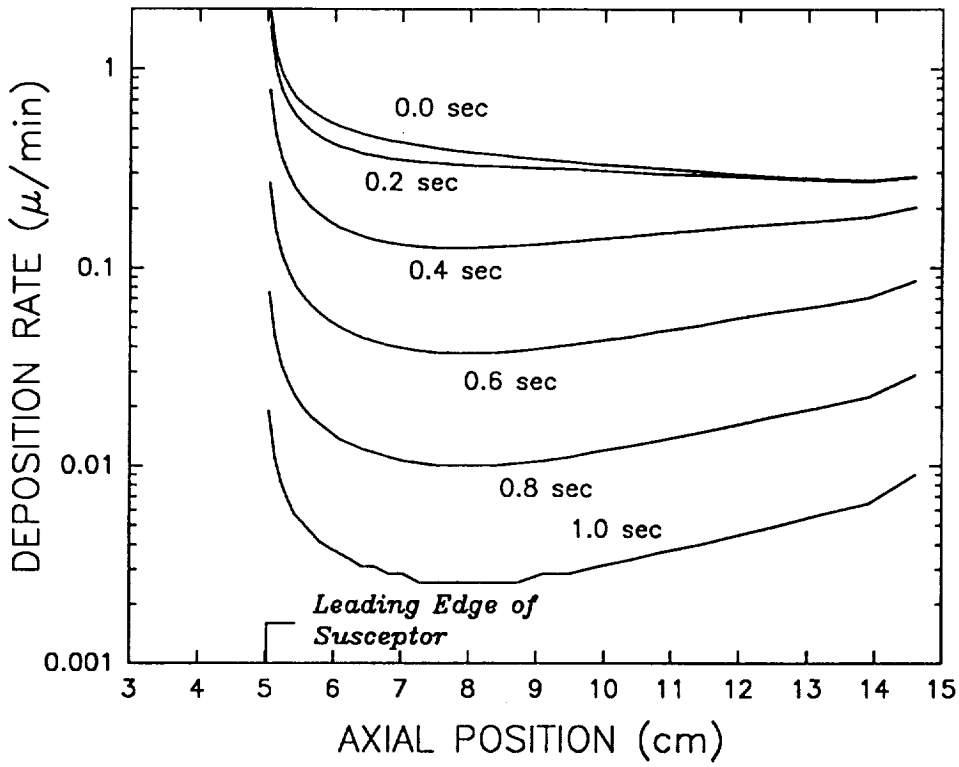


Figure 4.20: Deposition Rate for Transient Case G

## 4.8 CASE H – PARTICLE TRACKING AND THERMOPHORESIS

Case H illustrates particle tracking with and without thermophoresis. In particular, particle trajectories are calculated for the baseline simulation–Case A. Thus, the prevailing fluid flow pattern and temperature distribution are the same as described in Section 4.1 and shown in Figures 4.3 and 4.4.

Figure 4.21 shows how particle trajectories are affected by the diameter of the particle. The bottom plot shows particle trajectories without thermophoresis for particles of diameter 10, 100 and 1000 micron. The smallest particle of 10 micron diameter follows closely the fluid flow pattern; it rises at the leading edge of the susceptor where the gas begins to heat up and falls at the back edge of the susceptor where the gas expands into the larger flow channel. Particles of 100 micron diameter tend to follow the fluid flow but there is some relative motion between the particle and fluid. In this case, the inertia of the particle is becoming comparable to fluid drag forces. At 1000 micron diameter, the inertia of the particle is dominant and its trajectory follows a straight line unaffected by fluid drag forces.

The top plot of Figure 4.21 shows the particle trajectories under the same conditions except that thermophoretic forces are included. The 10 and 100 micron diameter particles are strongly affected by thermophoresis; these particles cross streamlines as they are transported

to the top cold wall of the reactor. The 1000 micron diameter particles, in contrast, follow a straight path. The impact of particle diameter on thermophoresis can be explained by reference to equation 3.91. (For the conditions of Case H, $Kn << 1$ and equation 3.91 applies rather than equation 3.92 for $Kn >> 1$.) According to equation 3.91, the thermophoretic force, $\boldsymbol{F}_t$, scales linearly with the particle diameter, $D_p$. The inertia force, however, scales with the particle mass or $D_p^3$. Thus, thermophoresis becomes less important as the particle diameter increases; for Case H, this occurs between 100 and 1000 micro diameter.

The particle trajectories shown in Figure 4.22 show how the point of origination of the particle affects its subsequent trajectory. Thermophoresis is neglected in the bottom plot and is included in the top plot. For both plots, 10 micron diameter particles originate from three positions at the reactor inlet: 1/4, 1/2 and 3/4 of the distance from the bottom wall to the top wall. Without thermophoresis, the 10 micron diameter particles follow the fluid flow pattern as was observed also in Figure 4.21. With thermophoresis, however, the particles are deflected toward the cold wall. Further, particles originating from about the top half of the inlet are predicted to strike the top wall while those which originate from the lower half will escape from the reactor.

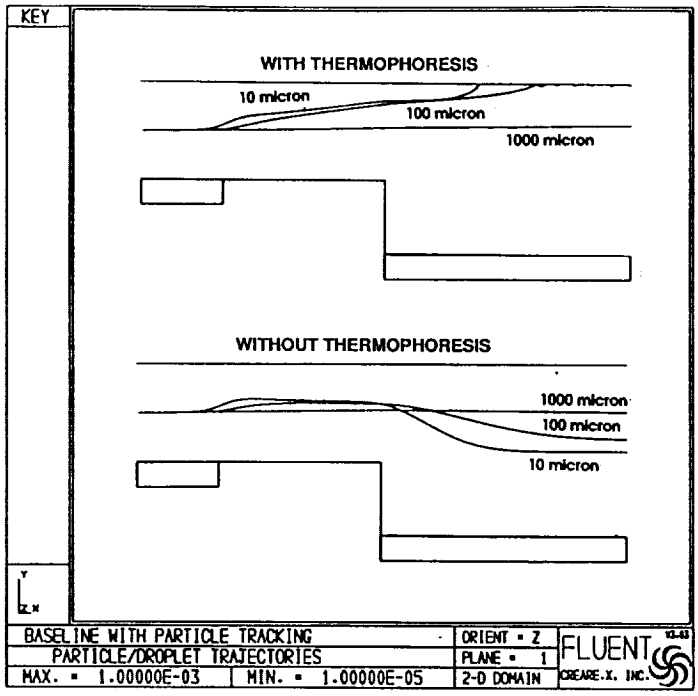


Figure 4.21: Effect of Diameter Thermophoresis On Particles

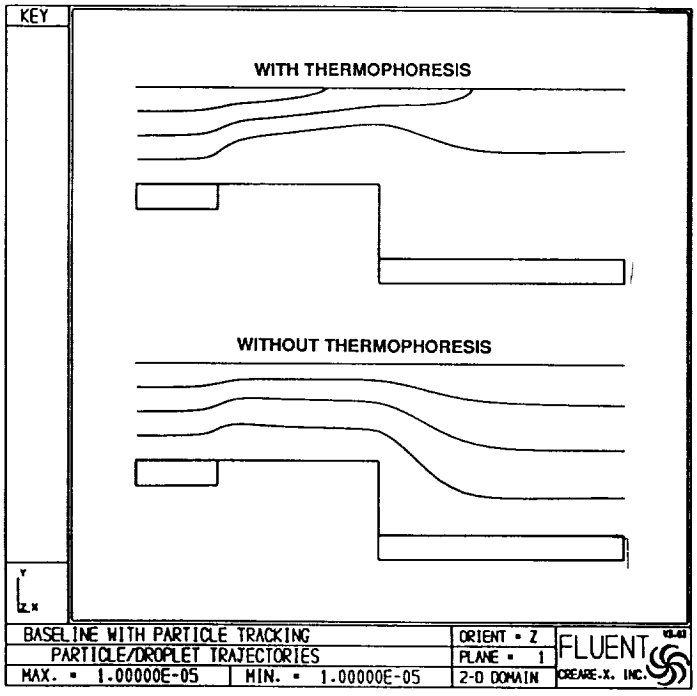


Figure 4.22: Effect of Point of Origination on Thermophoresis of Particle

# 5 CONCLUSIONS

Signicant new numerial modeling tools for chemical vapor deposition (CVD) have been developed under this SBIR program. The new modeling capabilities have been incorporated into the commerical computational fluid dynamics (CFD) software FLUENT.[1] As opposed to developing all software from scratch, use of FLUENT has provided a direct path to Phase III commercialization of the SBIR project results. Most of the CVD modeling capabilities are already available to the marketplace in FLUENT Version 3.03; the complete set of capabilities will be available upon the upcoming release of FLUENT Version 4.2.

The modeling capabilities which were developed for FLUENT include:

- Species transport of an arbitrary number of chemical species
- An arbitrary number of chemical reactions
- Surface reactions
- Surface deposition/etching
- Thermal (Soret) diffusion
- Composition dependent fluid properties
- Fluid properties calculated via kinetic theory models
- User-defined reaction rate expressions
- Thermophoresis of particles
- Curvilinear geometry mapping
- Time-varying gravity field

With these modeling capabilities, numerical simulation can provide an effective approach to help understand and optimize complex CVD processes.

Prior to this project, numerical simulation of CVD followed two rather limited approaches. First, commercial CFD software could be used. But since many of the CVD

[1]FLUENT is a family of computational fluid dynamics software originally developed by Creare Inc. Presently it continues to be developed and is marketed by Fluent Inc.

phenomena could not be simulated by any of the commercial codes, the researcher was restricted to deducing results from the information which could be obtained such as fluid flow patterns, or making analogies between thermal and chemical-species patterns. Second, software with the required CVD models could be developed from scratch. The resultant software, however, was never general purpose in scope and, of course, resources were needed to develop and debug the software. This project, especially with the early Phase III commercialization, represented a maturation of CFD for CVD. That is, many important phenomena of CVD can be routinely simulated in the same way that other fluid equipment (e.g., heat exchangers and turbines) have been simulated for years. This maturation is further evidenced by the availability of CVD simulation capabilities in other commercial software since the FLUENT capabilities were released.

As a result of the maturation of CFD for CVD, many of the numerical modeling studies of CVD recently published could have been undertaken with this commercial software with much less effort. The CVD scientist no longer needs to direct attention to numerical methods to achieve greater understanding of the CVD process. Evidence for this conclusion is underscored by recent project work at Creare which has utilized FLUENT to analyze CVD and etching reactors for equipment vendors in order to improve their equipment and for process engineers in order to optimize performance within a given reactor configuration. Additionally, several CVD equipment manufaturers and users have obtained licenses of FLUENT to perform their analysis in-house.

The modeling capabilities now available for CVD simulation are appreciable, but more can be done to make the modeling process more efficient. In particular, the following improvements would provide practical benefit for CVD simulation.

1. Speed of computation, especially when using the full multicomponent species transport model, as opposed to the dilute species transport model, and for large three-dimensional models.

   The inevitable increase in computational speed of computers at reduced costs will directly benefit CVD modeling. What is impractical to simulate today will become routine just several years from now.

2. Robust convergence algorithms for complex reaction mechanisms.

   Solutions to models with complex reaction mechanisms are computationally intensive at best, and sometimes solutions cannot be achieved at all. Improved algorithms are needed to handle many reactions occurring on vastly different time scales. This is an active area of current research in CFD; as improved techniques become available, it is expected that they will be readily implemented in the commercial software and become available for CVD modeling.

# Bibliography


[1] Paynter, G.C., Forester, C.K., and Tjonneland, E.; *CFD for Engine-Airframe Integration*; J. of Engrg. for Gas Turbine and Power, Vol. 109, April 1987.

[2] Jasinski, T.J., Sheikholeslami, M.Z., and Fretz, K.W.; *Numerical Modeling Tools for Chemical Vapor Deposition*, Phase I Final Report; Creare TM-1178, Creare Inc., Hanover, NH, August 28, 1987.

[3] FLUENT Version 3.03 User's Manual; Fluent Inc., Lebanon, NH, March 5, 1990.

[4] Merk, H.J.; *The Macroscopic Equations for Simultaneous Heat and Mass Transfer in Isotropic, Continuous and Closed Systems*; Appl. Sci. Res., Section A, Vol. 8, 1958, pp. 73-99.

[5] Pao, R.H.F.; Fluid Dynamics; Merrill Books, Columbus, OH, 1967.

[6] White, F.M.; Viscous Fluid Flow; McGraw-Hill, New York, NY, 1974.

[7] Patankar, S.V.; Numerical Heat Transfer and Fluid Flow, McGraw-Hill, New York, NY, 1980.

[8] Jensen, K.F., and Graves, D.B.; *Modeling and Analysis of Low Pressure CVD Reactors*; J. Electrochemical Society, Vol. 130, No. 9, 1983, pp. 1950-1957.

[9] Bird, R.B., Stewart, W.E. and Lightfoot, E.N.; Transport Phenomena, Wiley, New York, NY, 1960.

[10] Hirschfelder, J.O., Curtiss, C.F., and Bird, R.B.; Molecular Theory of Gases and Liquids; Wiley, New York, NY, 1964.

[11] Kuo, K.K.Y.; Principles of Combustion; Wiley, New York, NY, 1986.

[12] Reid, R.C., and Sherwood, T.K.; The Properties of Gases and Liquids; McGraw-Hill, New York, NY, 1966.

[13] Brodkey, R.S.; The Phenomena of Fluid Motions; Addison-Wesley, Reading, MA, 1967.

[14] Talbot, L., et. al.; *Thermophoresis of Particles In a Heated Boundary Layer*; J. Fluid Mech., Vol. 101, Part 4, 1980, pp. 737-758.

[15] Coltrin, M.E., Kee, R.J., and Miller, J.A.; *A Mathematical Model of Silicon Chemical Vapor Deposition. Further Refinements and the Effects of Thermal Diffusion*; J. Electrochemical Soc., Vol. 133, No. 6, June 1986, pp. 1206-1213.

[16] Gökoğlu, S.A.; *Significance of Vapor Phase Chemical Reactions on CVD Rates Predicted by Chemically Frozen and Local Thermodynamical Equilibrium Boundary Layer Theories*; J. Electrochem. Soc., Vol. 135, No. 6, June 1988, pp. 1562-1570.

# A DILUTE APPROXIMATION FOR MASS FLUX

In this appendix, we develop a simplified expression for the mass flux of species $i$, $\mathbf{J_i}$, in an ideal gas mixture of N species when all species except species N are present in dilute concentrations.

$$Y_i << Y_N .....i \neq N \tag{A.1}$$

The general expression for mass flux in an ideal gas mixture is (c.f., equation 3.19)

$$\mathbf{J_i} = \frac{C^2}{\rho} \sum_{k=1,k\neq i}^{N} \mathcal{M}_i \mathcal{M}_k D_{ik} \nabla Y_k - D_i^T \frac{\nabla T}{T} \tag{A.2}$$

In particular, simplified expressions for $D_{ik}$ and $\nabla Y_k$ in equation A.2 will be developed.

Using equation 3.21, the gradient of the mole fraction can be expanded as follows:

$$\nabla Y_k = \left[ \frac{\nabla X_k}{\mathcal{M}_k} \frac{1}{\sum_{i=1}^{N} \frac{X_i}{\mathcal{M}_i}} - \frac{X_k}{\mathcal{M}_k} \nabla \left( \frac{1}{\sum_{i=1}^{N} \frac{X_i}{\mathcal{M}_i}} \right) \right] \tag{A.3}$$

Multiplying equation A.1 by $M^{-1}$ and using equation 3.20, yields

$$\frac{Y_i}{\mathcal{M}} = \frac{X_i}{\mathcal{M}_i} << \frac{X_N}{\mathcal{M}_N} = \frac{Y_N}{\mathcal{M}} \tag{A.4}$$

Using equation A.4 simplifies the summations in equation A.3:

$$\sum_{i=1}^{N} \frac{X_i}{\mathcal{M}_i} = \frac{X_N}{\mathcal{M}_N} \tag{A.5}$$

The expression for the average molecular weight, equation 3.22, simplifies with equation A.5 to the following:

$$M = \frac{\rho}{C} = \mathcal{M}_N \tag{A.6}$$

Equation A.6 also assumes that $X_N \simeq 1$. Equation A.3 readily simplifies by substitution of equations A.5 and A.6:

$$\nabla Y_k = \frac{\mathcal{M}_N}{\mathcal{M}_k} \nabla X_k \tag{A.7}$$

Next the expression for $D_{ij}$ is simplified. The general expression for $D_{ij}$, given in equation 3.81, is repeated below:

$$D_{ij} = \frac{\mathcal{M}}{\mathcal{M}_j} \frac{K^{ji} - K^{ii}}{|K|} \tag{A.8}$$

where $|K|$ is the determinant of a matrix K and $K^{ji}$ and $K^{ii}$ are cofactors of $K_{ij}$. The elements of matrix K, $K_{ij}$, are given by equations 3.82 and 3.83. Using the dilute approximation, equation A.1, the values of $K_{ij}$ become:

$$K_{ij}(i \neq N) = \frac{\mathcal{M}_j}{\mathcal{M}_i} \frac{1}{D_{iN}}$$

$$K_{Nj}..... = \frac{1}{D_{Nj}} \tag{A.9}$$

$$K_{ii}... = 0$$

A somewhat tedious exercise in matrix algebra shows that

$$|K| = \frac{(-1)^{N+1} \sum_{m=1}^{N-1} \frac{\mathcal{M}_N}{\mathcal{M}_m D_{mN}}}{\prod_{m=1}^{N-1} D_{mN}} \tag{A.10}$$

$$K^{ii} = \frac{(-1)^{N} \sum_{m=1,m\neq i}^{N-1} \frac{\mathcal{M}_N}{\mathcal{M}_m D_{mN}}}{\prod_{m=1,m\neq i}^{N-1} D_{mN}} \tag{A.11}$$

$$K^{ji} = \frac{(-1)^{N+1} \frac{\mathcal{M}_N}{\mathcal{M}_i D_{iN}}}{\prod_{m=1,m\neq i}^{N-1} D_{mN}} \tag{A.12}$$

The simplified expression for $D_{ij}$ is obtained by substituting equations A.10, A.11, and A.12 into equation A.8

$$D_{ij} = \frac{\mathcal{M}}{\mathcal{M}_j} D_{iN}.....(i \neq j)$$

$$D_{ij} = 0.....(i = j) \tag{A.13}$$

The final expression for $\mathbf{J_i}$ in a dilute mixture is obtained by substituting equations A.6, A.7 and A.13 into equation A.2 and simplifying:

$$\mathbf{J_i} = \frac{\rho}{\mathcal{M}_N^2} \sum_{k=1,k\neq i}^{N} \mathcal{M}_i \mathcal{M}_k \left( \frac{\mathcal{M}_N}{\mathcal{M}_k} D_{iN} \right) \nabla Y_k - D_i^T \frac{\nabla T}{T}$$

$$\mathbf{J_i} = \rho \frac{\mathcal{M}_i}{\mathcal{M}_N} D_{iN} \sum_{k=1, k \neq i}^{N} \nabla Y_k - D_j^T \frac{\nabla T}{T}$$

$$\mathbf{J_i} = -\rho D_{iN} \frac{\mathcal{M}_i}{\mathcal{M}_N} \nabla Y_i - D_i^T \frac{\nabla T}{T}$$

$$\mathbf{J_i} = -\rho D_{iN} \nabla X_i - D_i^T \frac{\nabla T}{T} \tag{A.14}$$

Equation A.14 is used to evaluate the mass flux of species $i$ when the user chooses the dilute mixture approximation.

# B EXAMPLE OF FORTRAN SUBROUTINE FOR USER-DEFINED REACTION RATES

The following pages provide a listing of the Fortran source code subroutines which implement the user-defined reaction rate expressions given by equations 3.49a and 3.50a. Default versions of these subroutines are provided with FLUENT V3.03 as part of the user subroutine option package.

The first subroutine shown, USRRAT, can control access to multiple user-defined reaction rate expressions. The control is provided by the parameter UINDEX which is a menu prompted input. The Jensen and Graves [8] reaction rate expression is accessed by UINDEX=2 which transfers execution to another subroutine, JENGRV. JENGRV is the second subroutine shown. It is within JENGRV that equations 3.49a and 3.50a are evaluated. Two other reaction rate expressions, COLTRN (UINDEX=1) and HC4STG (UINDEX=3) are also accessed via USRRAT.

More information regarding development and use of user defined subroutines is provided in standard FLUENT documentation.

```
      SUBROUTINE USRRAT(  VIP,    VIDP,
     +                    SARRAY, MOLWT,
     +                    UINDEX, TYPE,   GASCON,
     +                    APREXP, ENGACT, BETA,   NUJ,
     +                    PRESS,  TEMP,   DENSTY,
     +                    RAY,    EPSILN, AMIX,   BMIX,
     +                    REALST, PROLST, NREACS, NPRODS,
     +                    NS,     NR,     NSPGAS, NOSPEC,
     +                    A,      B                       )
C SCCS ID @(#)usrrat.F 4.3 4/10/91
C-----------------------------------------------------------------------C
C                                                                       C
C     NAME : USRRAT                                                     C
C                                                                       C
C     PROGRAM : FLUENT                                                  C
C     VERSION : V3.00                                                   C
C                                                                       C
C     ARGS                                                              C
C                                                                       C
C       INPUT :  VIP    - MOLAR STOICH. COEFF. FOR REACTANTS            C
C                VIDP   - MOLAR STOICH. COEFF. FOR PRODUCTS             C
C                SARRAY - SPECIES CONCENTRATIONS                        C
C                MOLWT  - SPECIES MOLECULAR WEIGHT                      C
C                UINDEX - USER RATE EXPRESSION INDEX (THIS IS THE       C
C                         NUMBER INPUT IN THE RATE CONSTANTS TABLE)     C
C                TYPE   - REACTION TYPE FOR NR                          C
C                         = 0 - GAS PHASE                               C
C                         = 1 - SURFACE                                 C
C                GASCON - GAS CONSTANT                                  C
C                APREXP - PRE-EXPONENTIAL FACTOR                        C
C                ENGACT - ACTIVATION ENERGY                             C
C                BETA   - TEMPERATURE EXPONENT                          C
```

```
C              NUJ    - REACTANT SPECIES RATE EXPONENTS                C
C              PRESS  - ABSOLUTE PRESSURE                              C
C              TEMP   - TEMPERATURE                                    C
C              DENSTY - DENSITY                                        C
C              KAY    - TURBULENT KINETIC ENERGY                       C
C              EPSILN - DISSIPATION RATE OF TURBULENT K.E.             C
C              AMIX   - TURBULENT MIXING RATE CONSTANT - A             C
C              BMIX   - TURBULENT MIXING RATE CONSTANT - B             C
C              REALST - LIST OF REACTNAT SPECIES                       C
C              PROLST - LIST OF PRODUCT SPECIES                        C
C              NREACS - NUMBER OF REACTANT SPECIES                     C
C              NPRODS - NUMBER OF PRODUCT SPECIES                      C
C              NS     - SPECIES NUMBER FOR SPECIES S                   C
C              NR     - REACTION NUMBER                                C
C              NSPGAS - NUMBER OF GAS PHASE SPECIES                    C
C              NOSPEC - NUMBER OF SPECIES ACTIVE                       C
C                                                                      C
C     OUTPUT : A      - CONSTANT TERM IN RATE OF MASS ADDITION         C
C                       TO SPECIES S                                   C
C              B      - LINEAR TERM IN RATE OF MASS ADDITION           C
C                       TO SPECIES S                                   C
C                                                                      C
C  PURPOSE : THIS ROUTINE ALLOWS THE USER TO PROGRAM A SPECIFIC        C
C            RATE LAW BASED ON THE SPECIES CONCENTRATIONS, THE          C
C            THERMODYNAMIC PROPERTIES, TURBULENCE CHARACTERISTICS       C
C            AND STOICHIOMETRY.                                        C
C                                                                      C
C  COMMENTS : THE VARIABLES ARE ALL IN S.I. UNITS.  THE UNITS USED     C
C             ARE :                                                    C
C                                                                      C
C             VIP    - DIMENSIONLESS                                   C
C             VIDP   - DIMENSIONLESS                                   C
C             SARRAY - DIMENSIONLESS                                   C
C             MOLWT  - KG./KG.-MOLE                                    C
C             GASCON - JOULES/KG.-K                                    C
C             PRESS  - PASCALS                                         C
C             TEMP   - KELVIN                                          C
C             DENSTY - KG./CU.METER (RHO)                              C
C             KAY    - M.SQ./SEC. SQ.                                  C
C             EPSILN - M.SQ./SEC.**3                                   C
C             AMIX   - DIMENSIONLESS                                   C
C             BMIX   - DIMENSIONLESS                                   C
C                                                                      C
C             A AND B MUST HAVE UNITS TO BE CONSISTENT WITH THE        C
C             SPECIES CONSERVATION EQUATION, I.E. :                    C
C                                                                      C
C             D(RHO*S)                                                 C
C        VOLUME*--------  = (A + B*S)*VOLUME + ... (GAS PHASE)         C
C                DT                                                    C
C                                                                      C
C                          OR                                          C
C                                                                      C
C             D(RHO*S)                                                 C
C        VOLUME*--------  = (A + B*S)*AREA + ...   (SURFACE)           C
C                DT                                                    C
C                                                                      C
C                                                                      C
C                                                                      C
C             NOTE : NUMERICAL STABILITY REQUIRES THAT THE LINEAR      C
C                    COEFFICIENT, B, ALWAYS BE NEGATIVE.  THE          C
C                    ROUTINES WHICH USE A AND B WILL SET B TO ZERO     C
C                    IF IT IS POSITIVE.                                C
C                                                                      C
C             NOTE : THE PROGRAM WILL TERMINATE IF THE RATE            C
C                    EXPRESSIO HAS NOT BEEN IMPLEMENTED                C
C                                                                      C
C  CALLED BY : RATE                                                    C
C                                                                      C
```

```
C     SUBROUTINES CALLED : SEE LIST BELOW                                C
C                                                                        C
C     CREATED : 11/07/88                                                 C
C                                                                        C
C     AUTHOR  : K. W. FRETZ, CREARE, INC., HANOVER, NH                   C
C                                                                        C
C     REVISION HISTORY :                                                 C
C                                                                        C
C     NO.    BY     DATE            MODIFICATIONS/COMMENTS              C
C     ---    ---    --------   ----------------------------------------  C
C      01    JPM    12/04/90   ADDED EPC UPDATE #29.                    C
C                               REPLACED VARIABLE STRLEN WITH THE VALUE  C
C                               40 SO THAT THE MESSAGE AT THE END OF THE C
C                               SUBROUTINE GETS PRINTED.                 C
C      02    SDG    91/03/28   Converted all INTEGER strings to         C
C                               FORTRAN-77 CHARACTER data types.         C
C                                                                        C
C-------------------------------------------------------------------------C
C            RATE EXPRESSIONS CURRENTLY IMPLEMENTED                      C
C-------------------------------------------------------------------------C
C                                                                        C
C                                                                        C
C                                                          DATE          C
C   INDEX   SUBROUTINE NAME   DESCRIPTION/REFERENCE        ADDED     BY  C
C   -----   ---------------   ---------------------        --------  --- C
C     1        COLTRN         COLTRIN SILANE DECOMP.                    C
C                             TO SIH2/H2 DECOMP.          03/21/89  KWF C
C     2        JENGRV         JENSEN, GRAVES SILANE                     C
C                             TO SIH2/H2 DECOMP.          03/21/89  KWF C
C     3        HC4STG         4-STAGE HYDROCARBON                       C
C                             COMBUSTION                  03/21/89  KWF C
C                                                                        C
C-------------------------------------------------------------------------C
C
#include "IMPLICIT.INC"
C
C-------------------------------------------------------------------------C
C     ARGUMENT TYPE DECLARATIONS...                                      C
C-------------------------------------------------------------------------C
C
      INTEGER NSPGAS
      INTEGER NOSPEC
      INTEGER UINDEX
      INTEGER TYPE
      INTEGER NR
      INTEGER NS
      INTEGER NREACS
      INTEGER NPRODS
      INTEGER REALST(NOSPEC)
      INTEGER PROLST(NOSPEC)
C
      REAL    VIP(NOSPEC)
      REAL    VIDP(NOSPEC)
      REAL    SARRAY(NOSPEC)
      REAL    MOLWT(NOSPEC)
      REAL    APREXP
      REAL    ENGACT
      REAL    BETA
      REAL    NUJ(NOSPEC)
      REAL    GASCON
      REAL    PRESS
      REAL    TEMP
      REAL    DENSTY
      REAL    KAY
      REAL    EPSILN
      REAL    AMIX
      REAL    BMIX
      REAL    A
      REAL    B
```

```
C
C-----------------------------------------------------------------------C
C     LOCAL VARIABLE TYPE DECLARATIONS...                                C
C-----------------------------------------------------------------------C
C
      CHARACTER*40 STRING
C
C-----------------------------------------------------------------------C
C     BEGIN SUBROUTINE USRRAT...                                         C
C-----------------------------------------------------------------------C
C
        A = 0.0
        B = 0.0
C
        IF      ( UINDEX .EQ. 1 ) THEN
C
C=======================================================================C
C       COLTRIN, ET. AL. SILANE DECOMPOSTION...                          C
C=======================================================================C
C
            CALL COLTRN( VIP,    VIDP,
     +                   SARRAY, MOLWT,
     +                   UINDEX, TYPE,   GASCON,
     +                   APREXP, ENGACT, BETA,   NUJ,
     +                   PRESS,  TEMP,   DENSTY,
     +                   KAY,    EPSILN, AMIX,   BMIX,
     +                   REALST, PROLST, NREACS, NPRODS,
     +                   NS,     NR,     NSPGAS, NOSPEC,
     +                   A,      B                       )
C
        ELSE IF ( UINDEX .EQ. 2 ) THEN
C
C=======================================================================C
C       JENSEN AND GRAVES SILANE DECOMPOSTION...                         C
C=======================================================================C
C
            CALL JENGRV( VIP,    VIDP,
     +                   SARRAY, MOLWT,
     +                   UINDEX, TYPE,   GASCON,
     +                   APREXP, ENGACT, BETA,   NUJ,
     +                   PRESS,  TEMP,   DENSTY,
     +                   KAY,    EPSILN, AMIX,   BMIX,
     +                   REALST, PROLST, NREACS, NPRODS,
     +                   NS,     NR,     NSPGAS, NOSPEC,
     +                   A,      B                       )
C
        ELSE IF ( UINDEX .EQ. 3 ) THEN
C
C=======================================================================C
C       4-STAGE HYDROCARBON COMBUSTION...                                C
C=======================================================================C
C
            CALL HC4STG(  VIP,    VIDP,   SARRAY, MOLWT,
     +                    GASCON, TEMP,   DENSTY,
     +                    KAY,    EPSILN, NS,     NR,
     +                    NSPGAS, NOSPEC, A,      B      )
C
        ELSE
C
C=======================================================================C
C       USER RATE NOT FOUND...                                           C
C=======================================================================C
C
            WRITE(STRING,
     +          FMT='('' USER RATE NO. '''//
     +               ',I4,''// ''' IS NOT DEFINED'')') UINDEX
            CALL MESSGE( STRING, 40)
            STOP
```

```
C
        ENDIF
C
C-----------------------------------------------------------------------C
C     END SUBROUTINE USRRAT AND RETURN...                               C
C-----------------------------------------------------------------------C
C
      RETURN
      END
```

```
      SUBROUTINE JENGRV(  VIP,    VIDP,
     +                    SARRAY, MOLWT,
     +                    UINDEX, TYPE,   GASCON,
     +                    APREXP, ENGACT, BETA,   NUJ,
     +                    PRESS,  TEMP,   DENSTY,
     +                    KAY,    EPSILN, AMIX,   BMIX,
     +                    REALST, PROLST, NREACS, NPRODS,
     +                    NS,     NR,     NSPGAS, NOSPEC,
     +                    A,      B                       )
C SCCS ID @(#)jengrv.F 4.1 11/19/90
C-----------------------------------------------------------------------C
C                                                                       C
C     NAME : JENGRV                                                     C
C                                                                       C
C     PROGRAM : FLUENT                                                  C
C     VERSION : V3.00                                                   C
C                                                                       C
C     ARGS                                                              C
C                                                                       C
C       INPUT :  VIP    - MOLAR STOICH. COEFF. FOR REACTANTS            C
C                VIDP   - MOLAR STOICH. COEFF. FOR PRODUCTS             C
C                SARRAY - SPECIES CONCENTRATIONS                        C
C                MOLWT  - SPECIES MOLECULAR WEIGHT                      C
C                UINDEX - USER RATE EXPRESSION INDEX (THIS IS THE       C
C                         NUMBER INPUT IN THE RATE CONSTANTS TABLE)     C
C                TYPE   - REACTION TYPE FOR NR                          C
C                         = 0 - GAS PHASE                               C
C                         = 1 - SURFACE                                 C
C                GASCON - GAS CONSTANT                                  C
C                APREXP - PRE-EXPONENTIAL FACTOR                        C
C                ENGACT - ACTIVATION ENERGY                             C
C                BETA   - TEMPERATURE EXPONENT                          C
C                NUJ    - REACTANT SPECIES RATE EXPOENENTS              C
C                PRESS  - ABSOLUTE PRESSURE                             C
C                TEMP   - TEMPERATURE                                   C
C                DENSTY - DENSITY                                       C
C                KAY    - TURBULENT KINETIC ENERGY                      C
C                EPSILN - DISSIPATION RATE OF TURBULENT K.E.            C
C                AMIX   - TURBULENT MIXING RATE CONSTANT - A            C
C                BMIX   - TURBULENT MIXING RATE CONSTANT - B            C
C                REALST - LIST OF REACTNAT SPECIES                      C
C                PROLST - LIST OF PRODUCT SPECIES                       C
C                NREACS - NUMBER OF REACTANT SPECIES                    C
C                NPRODS - NUMBER OF PRODUCT SPECIES                     C
C                NS     - SPECIES NUMBER FOR SPECIES S                  C
C                NR     - REACTION NUMBER                               C
C                NSPGAS - NUMBER OF GAS PHASE SPECIES                   C
C                NOSPEC - NUMBER OF SPECIES ACTIVE                      C
C                                                                       C
C        OUTPUT : A      - CONSTANT TERM IN RATE OF MASS ADDITION       C
C                         TO SPECIES S                                  C
C                B      - LINEAR TERM IN RATE OF MASS ADDITION         C
C                         TO SPECIES S                                  C
C                                                                       C
C     PURPOSE : THIS ROUTINE OMPLEMENTS THE LOW PRESSURE, JENSEN AND    C
C               GRAVES RATE EXPERSSION FOR SILANE DECOMPOSTION.         C
C                                                                       C
C     COMMENTS : THE VARIABLES ARE ALL IN S.I. UNITS.  THE UNITS USED   C
C                ARE :                                                  C
C                                                                       C
C                VIP    - DIMENSIONLESS                                 C
C                VIDP   - DIMENSIONLESS                                 C
C                SARRAY - DIMENSIONLESS                                 C
C                MOLWT  - KG./KG.-MOLE                                  C
C                GASCON - JOULES/KG.-K                                  C
C                PRESS  - PASCALS                                       C
C                TEMP   - KELVIN                                        C
C                DENSTY - KG./CU.METER (RHO)                            C
```

```
C               KAY    - M.SQ./SEC. SQ.                                 C
C               EPSILM - M.SQ./SEC.**3                                  C
C               AMIX   - DIMENSIONLESS                                  C
C               BMIX   - DIMENSIONLESS                                  C
C                                                                       C
C               A AND B MUST HAVE UNITS TO BE CONSISTENT WITH THE       C
C               SPECIES CONSERVATION EQUATION, I.E. :                   C
C                                                                       C
C               D(RHO*S)                                                C
C         VOLUME*--------  = (A + B*S)*VOLUME + ... (GAS PHASE)         C
C                  DT                                                   C
C                                                                       C
C                           OR                                          C
C                                                                       C
C               D(RHO*S)                                                C
C         VOLUME*--------  = (A + B*S)*AREA + ...   (SURFACE)           C
C                  DT                                                   C
C                                                                       C
C                                                                       C
C                                                                       C
C               NOTE : NUMERICAL STABILITY REQUIRES THAT THE LINEAR     C
C                      COEFFICIENT, B, ALWAYS BE NEGATIVE.  THE         C
C                      ROUTINES WHICH USE A AND B WILL SET B TO ZERO    C
C                      IF IT IS POSITIVE.                               C
C                                                                       C
C               THE SPECIES NUMBERS ARE AS FOLLOWS :                    C
C                                                                       C
C                   SIH4 = 1                                            C
C                   H2   = 2                                            C
C                   HE   = 3 (HELUIM CARRIER, NOT USED FOR HYDROGEN     C
C                             CARRIER                                   C
C                   SI   = NOSPEC (LAST SPECIES)                        C
C                                                                       C
C     CALLED BY : USRRAT                                                C
C                                                                       C
C     SUBROUTINES CALLED : NONE                                         C
C                                                                       C
C     CREATED : 03/21/89                                                C
C                                                                       C
C     AUTHOR  : K. W. FRETZ, CREARE, INC., HANOVER, NH                  C
C                                                                       C
C-----------------------------------------------------------------------C
C
#include "IMPLICIT.INC"
C
C-----------------------------------------------------------------------C
C     ARGUMENT TYPE DECLARATIONS...                                     C
C-----------------------------------------------------------------------C
C
      INTEGER NSPGAS
      INTEGER NOSPEC
      INTEGER UINDEX
      INTEGER TYPE
      INTEGER NR
      INTEGER NS
      INTEGER NREACS
      INTEGER NPRODS
      INTEGER REALST(NOSPEC)
      INTEGER PROLST(NOSPEC)
C
      REAL    VIP(NOSPEC)
      REAL    VIDP(NOSPEC)
      REAL    SARRAY(NOSPEC)
      REAL    MOLWT(NOSPEC)
      REAL    APREXP
      REAL    ENGACT
      REAL    BETA
      REAL    MUJ(NOSPEC)
```

```
      REAL    GASCON
      REAL    PRESS
      REAL    TEMP
      REAL    DENSTY
      REAL    KAY
      REAL    EPSILN
      REAL    AMIX
      REAL    BMIX
      REAL    A
      REAL    B
C
C-----------------------------------------------------------------------C
C     LOCAL VARIABLE TYPE DECLARATIONS...                               C
C-----------------------------------------------------------------------C
C
      INTEGER I
C
      INTEGER SIH4
      INTEGER SIL
      INTEGER H2
      INTEGER HE
C
      REAL    DENOM
      REAL    GAMMA(100)
      REAL    K
      REAL    K1
      REAL    K2
      REAL    PATOAT
      REAL    T
C
      PARAMETER ( K1     = 1.7300E+03,
     +            K2     = 4.0000E+04,
     +            PATOAT = 9.8692E-06  )
C
C-----------------------------------------------------------------------C
C     STATEMENT FUNCTION...                                             C
C-----------------------------------------------------------------------C
C
      K(T) = 0.028*1.25E+09*EXP( -18500.0/T )
C
C-----------------------------------------------------------------------C
C     BEGIN SUBROUTINE JENGRV...                                        C
C-----------------------------------------------------------------------C
C
        SIH4 = 1
        H2   = 2
        IF ( NSPGAS .EQ. 3 ) THEN
          HE = 3
        ELSE
          HE = -1
        ENDIF
        SIL  = NOSPEC
C
        DO 10 I = 1, NOSPEC
          GAMMA( I ) = DENSTY*TEMP*GASCON/MOLWT( I )*PATOAT
10      CONTINUE
C
        DENOM = 1.0 + K1*GAMMA(  H2  )*SARRAY(  H2  ) +
     +                K2*GAMMA( SIH4 )*SARRAY( SIH4 )
C
        IF      ( NS .EQ. SIH4 ) THEN
C
          A = 0.0
          B = -MOLWT( SIH4 )/MOLWT( SIL )*K( TEMP )*GAMMA( SIH4 )/DENOM
C
        ELSE IF ( NS .EQ. SIL  ) THEN
C
          A = K( TEMP )*GAMMA( SIH4 )*SARRAY( SIH4 )/DENOM
```

```
          B = 0.0
C
        ELSE IF ( MS .EQ. H2 ) THEN
C
          A = MOLWT( H2 )/MOLWT( SIL )*2.0*
     +        K( TEMP )*GAMMA( SIH4 )*SARRAY( SIH4 )/DENOM
          B = 0.0
C
        ELSE IF ( MS .EQ. HE ) THEN
C
          A = 0.0
          B = 0.0
C
        ENDIF
C
C-------------------------------------------------------------------------C
C     END SUBROUTINE JENGRV AND RETURN...                                 C
C-------------------------------------------------------------------------C
C
      RETURN
      END
```

# C DEFAULT EXPRESSIONS FOR FLUID PROPERTIES

The default expressions used by FLUENT to calculate fluid properties are summarized in the tables of this Appendix. See Section 3.4.2 for a more complete description of fluid properties.

Table C.1: **Nomenclature For Table C.2 - C.8**

| | |
|---|---|
| $C_p$ | specific heat |
| $\mathcal{D}_{ij}$ | binary diffusion coefficient |
| $D_i^T$ | thermal diffusion coefficient |
| $f$ | degrees of freedom of molecule |
| $i$ | species number |
| $j$ | species number |
| $k$ | thermal conductivity |
| $K$ | matrix defined by equations 3.82 and 3.83 |
| $K^{ii}$ | cofactor of matrix $K$ |
| $K^{ij}$ | cofactor of matrix $K$ |
| $\|K\|$ | determinant of matrix $K$ |
| $M$ | molecular weight |
| $n$ | total number of species |
| $P$ | pressure |
| $R$ | gas constant |
| $T$ | temperature |
| $X$ | mass fraction |
| $Y$ | mole fraction |
| GREEK | |
| $\epsilon/\kappa$ | potential energy well depth for the Lennard-Jones potential energy function |
| $\mu$ | viscosity |
| $\rho$ | density |
| $\sigma$ | molecule size parameter for the Lennard-Jones potential energy function |
| $\phi$ | function defined by equation 3.66 |
| $\Omega_{\mathcal{D}}$ | function for perdiction of binary diffusivity according to equation 3.75. Tabulated in reference[9]. |
| $\Omega_\mu$ | function for prediction of viscosity according to equation 3.75. Tabulated in reference [9]. |
| SUBSCRIPTS AND SUPERSCRIPTS | |
| $i$ | species number |
| $j$ | species number |
| $mix$ | gas mixture |

Table C.2: **Summary of Default Expressions for Density**

| Ideal Gas Model | Composition Independent | Composition Dependent | |
|---|---|---|---|
| | | Menu-Accessed | User-Subroutine |
| Enabled | $\frac{P\mathcal{M}}{RT}$ | $\frac{P}{RT\sum_{i=1}^{n}\frac{X_i}{\mathcal{M}_i}}$ | $\left[\sum_{i=1}^{n}\frac{X_i}{\rho_i}\right]^{-1}$ |
| Disabled | Polynomial Function of Temperature | | |
| | Piecewise Linear Function of Temperature | $\sum_{i=1}^{n}\rho_i X_i$ | |

Table C.3: **Summary of Default Expressions for Viscosity**

| Ideal Gas Model | Composition Independent | Composition Dependent | |
|---|---|---|---|
| | | Menu-Accessed | User-Subroutine |
| Enabled | $2.67(10^{-6})\frac{\sqrt{\mathcal{M}_i T}}{\sigma_i^2\Omega_\mu}$ | $\sum_{i=1}^{n}\frac{Y_i\mu_i}{\sum_{j=1}^{n}Y_j\phi_{ij}}$ | $\frac{\sum_{i=1}^{n}\sqrt{\mathcal{M}_i}\,Y_i\mu_i}{\sum_{i=1}^{n}\sqrt{\mathcal{M}_i}\,Y_i}$ |
| | Polynomial Function of Temperature | | |
| | Piecewise-Linear Function of Temperature | | |
| Disabled | Polynomial Function of Temperature | $\sum_{i=1}^{n}X_i\mu_i$ | |
| | Piecewise-Linear Function of Temperature | | |

$$\phi_{ij} = \frac{\left[1+\left[\frac{\mu_i}{\mu_j}\right]^{\frac{1}{2}}\left[\frac{\mathcal{M}_j}{\mathcal{M}_i}\right]^{\frac{1}{4}}\right]^2}{\sqrt{8(1+\mathcal{M}_i/\mathcal{M}_j)}}$$

Table C.4: **Summary of Default Expressions for Specific Heat**

| Ideal Gas Model | Composition Independent | Composition Dependent: Menu-Accessed | Composition Dependent: User-Subroutine |
|---|---|---|---|
| Enabled | $\frac{1}{2}\frac{R}{\mathcal{M}_i}(f_i+2)$ | $\sum_{i=1}^{n} X_i C_{p,i}$ | $\sum_{i=1}^{n} X_i C_{p,i}$ |
| | Polynomial Function of Temperature | | |
| | Piecewise-Linear Function of Temperature | | |
| Disabled | Polynomial Function of Temperature | $\sum_{i=1}^{n} X_i C_{p,i}$ | |
| | Piecewise-Linear Function of Temperature | | |

Table C.5: **Summary of Default Expressions for Thermal Conductivity**

| Ideal Gas Model | Composition Independent | Composition Dependent: Menu-Accessed | Composition Dependent: User-Subroutine |
|---|---|---|---|
| Enabled | $\mu_i \frac{15}{4}\frac{R}{\mathcal{M}_i}\left[\frac{4}{15}\frac{C_{p,i}\mathcal{M}_i}{R}+\frac{1}{3}\right]$ | $\sum_{i=1}^{n} \frac{Y_i k_i}{\sum_{j=1}^{n} Y_j \phi_{ij}}$ | $\frac{\sum_{i=1}^{n} 3\sqrt{\mathcal{M}_i}\, Y_i\, k_i}{\sum_{i=1}^{n} 3\sqrt{\mathcal{M}_i}\, Y_i}$ |
| | Polynomial Function of Temperature | | |
| | Piecewise-Linear Function of Temperature | | |
| Disabled | Polynomial Function of Temperature | $\sum_{i=1}^{n} X_i\, k_i$ | |
| | Piecewise-Linear Function of Temperature | | |

$$\phi_{ij} = \frac{\left[1+\left[\frac{\mu_i}{\mu_j}\right]^{\frac{1}{2}}\left[\frac{\mathcal{M}_j}{\mathcal{M}_i}\right]^{\frac{1}{4}}\right]^2}{\sqrt{8(1+\mathcal{M}_i/\mathcal{M}_j)}}$$

Table C.6: **Summary of Default Expressions for Binary Mass Diffusivity**

| Ideal Gas Model | Composition Independent | Composition Dependent: Menu-Accessed | Composition Dependent: User-Subroutine |
|---|---|---|---|
| Enabled | $\frac{0.0188}{P\sigma_{ij}^2\,\Omega_D}\sqrt{T^3\left[\frac{1}{\mathcal{M}_i}+\frac{1}{\mathcal{M}_j}\right]}$ | $\frac{1-Y_i}{\sum_{j=1,\neq i}^{n}\frac{Y_j}{\mathcal{D}_{ij}}}$ | Not Available |
| | Polynomial Function of Temperature | | |
| | Piecewise-Linear Function of Temperature | | |
| Disabled | Polynomial Function of Temperature | | |
| | Piecewise-Linear Function of Temperature | $\frac{1-Y_i}{\sum_{i=1,\neq i}^{n}\frac{Y_j}{\mathcal{D}_{ij}}}$ | |

$$\sigma_{ij} = \frac{1}{2}(\sigma_i + \sigma_j)$$

$$(\epsilon/\kappa)_{ij} = \sqrt{(\epsilon/\kappa)_i(\epsilon/\kappa)_j}$$

Table C.7: **Summary of Default Expressions for Multicomponent Mass Diffusivity**

| Ideal Gas Model | Composition Independent | Composition Dependent: Menu-Accessed | Composition Dependent: User-Subroutine |
|---|---|---|---|
| Enabled | Polynomial Function of Temperature | | |
| | Piecewise-Linear Function of Temperature | $\frac{\mathcal{M}_{mix}}{\mathcal{M}_i}\,\frac{K^{ji}-K^{ii}}{\lvert K\rvert}$ | No Default Expression Implemented |
| Disabled | Polynomial Function of Temperature | No Default Expression Implemented | |
| | Piecewise-Linear Function of Temperature | | |

$|K|$ = determinant of matrix $K$
$K^{ji}, K^{ii}$ = cofactors of matrix $K$
$K_{ij}$ = element $(i,j)$ of matrix $K$, defined by:

$$K_{ij} = \frac{Y_i}{\mathcal{D}_{ij}} + \frac{\mathcal{M}_j}{\mathcal{M}_i}\sum_{k=1,\neq i}^{n}\frac{Y_k}{\mathcal{D}_{ij}} \text{ for } i \neq j$$

$$K_{ij} = 0 \text{ for } i = j$$



C-2

Table C.8: **Summary of Default Expressions for Soret Coefficient**

| | | Composition Dependent | |
|---|---|---|---|
| **Ideal Gas Model** | **Composition Independent** | **Menu–Accessed** | **User-Subroutine** |
| Enabled | Polynomial Function of Temperature | $D_i^T = -2.59(10^{-7})T^{0.659}\left[\frac{\mathcal{M}_i^{0.511}Y_i}{\sum_{i=1}^{n}\mathcal{M}_i^{0.511}Y_i} - X_i\right]\left[\frac{\sum_{i=1}^{n}\mathcal{M}_i^{0.511}Y_i}{\sum_{i=1}^{n}\mathcal{M}_i^{0.489}Y_i}\right]$ | $D_i^T = -2.59(10^{-7})T^{0.659}\left[\frac{\mathcal{M}_i^{0.511}Y_i}{\sum_{i=1}^{n}\mathcal{M}_i^{0.511}Y_i} - X_i\right]\left[\frac{\sum_{i=1}^{n}\mathcal{M}_i^{0.511}Y_i}{\sum_{i=1}^{n}\mathcal{M}_i^{0.489}Y_i}\right]$ |
| | Piecewise-Linear Function of Temperature | | |
| Disabled | Polynomial Function of Temperature | Not Available | |
| | Piecewise-Linear Function of Temperature | | |

# REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188



| 1. AGENCY USE ONLY (Leave blank) | 2. REPORT DATE | 3. REPORT TYPE AND DATES COVERED |
|---|---|---|
| | December 1992 | Contractor Report (5/5/88-5/31/91) |

**4. TITLE AND SUBTITLE**
Numerical Modeling Tools for Chemical Vapor Deposition

**5. FUNDING NUMBERS**
C NAS1-18648
WU 324-02-00

**6. AUTHOR(S)**
Thomas J. Jasinski
Edward P. Childs

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**
Creare Inc.
P.O. Box 71
Hanover, NH 03755

**8. PERFORMING ORGANIZATION REPORT NUMBER**
TM-1504

**9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)**
National Aeronautics and Space Administration
Langley Research Center
Hampton, VA 23681-0001

**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**
NASA CR-4480

**11. SUPPLEMENTARY NOTES**
FLUENT is developed and marketed by Fluent Inc.
Edward P. Childs is presently with Fluent Inc., Lebanon, NH.
Langley Technical Monitor: Ivan O. Clark
Final Report

**12a. DISTRIBUTION / AVAILABILITY STATEMENT**
Unclassified-Unlimited

Subject Category 34

**12b. DISTRIBUTION CODE**

**13. ABSTRACT** (Maximum 200 words)

Development of general numerical simulation tools for chemical vapor deposition (CVD) was the objective of this investigation. Physical models of important CVD phenomena were developed and implemented into the commercial computational fluid dynamics software FLUENT (see block 11). The resulting software can address general geometries as well as the most important phenomena occurring within CVD reactors: fluid flow patterns, temperature and chemical species distribution, gas phase and surface deposition. This report documents the physical models which are available and provides examples of CVD simulation capabilities.

| 14. SUBJECT TERMS | 15. NUMBER OF PAGES |
|---|---|
| Chemical Vapor Deposition, CVD, CFD, Numerical Modeling, Computational Fluid Dynamics | 100 |
| | 16. PRICE CODE: A05 |

| 17. SECURITY CLASSIFICATION OF REPORT | 18. SECURITY CLASSIFICATION OF THIS PAGE | 19. SECURITY CLASSIFICATION OF ABSTRACT | 20. LIMITATION OF ABSTRACT |
|---|---|---|---|
| Unclassified | Unclassified | | |

NSN 7540-01-280-5500

Standard Form 298 (Rev 2-89)
Prescribed by ANSI Std Z39-18
298-102

NASA-Langley, 1992